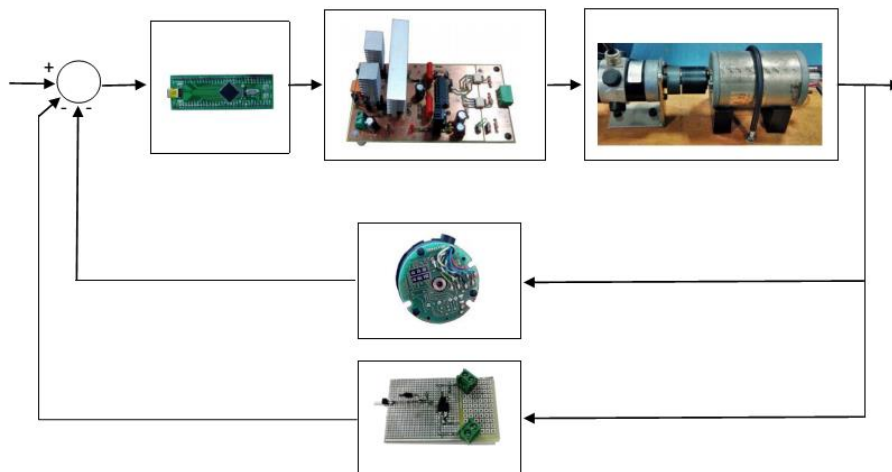




**Cláudio Filipe
Miranda
Dos Santos**

Controlo de Velocidade e Posição de um Motor DC por Realimentação de Estados





**Cláudio Filipe
Miranda
Dos Santos**

**Controlo de Velocidade e Posição de
um Motor DC por Realimentação de Estados**

Dissertação apresentada à Universidade de Aveiro para cumprimento dos requisitos necessários à obtenção do grau de Mestre em Engenharia Electrónica e Telecomunicações, realizada sob a orientação científica do Professor Doutor Alexandre Manuel Moutela Nunes da Mota, Professor Associado do Departamento de Eletrónica, Telecomunicações e Informática da Universidade de Aveiro e do Mestre António Luís Tenedório Pereira da empresa Globaltronic, Eletrónica e Telecomunicações, SA.

O júri / The jury

Presidente / President

Professor Doutor Paulo Bacelar Reis Pedreiras
Professor Auxiliar da Universidade de Aveiro

Vogais / Examiners Committee

Professor Doutor José António Barros Vieira
Professor Adjunto do Instituto Politécnico de Castelo Branco

Mestre António Luís Tenedório Pereira
Engenharia Eletrónica e Telecomunicações pela Universidade de Aveiro
(Co-orientador)

Agradecimentos / Acknowledgements

Este espaço é dedicado a todos aqueles que deram a sua contribuição para que este projeto fosse realizado.

Quero deixar uma palavra de agradecimento ao meu orientador Prof. Dr. Alexandre Mota pela excelente orientação, conselhos, críticas construtivas e disponibilidade que revelou nas várias fases do trabalho.

Ao meu co-orientador Mestre Luís Tenedório por todo o apoio científico e técnico prestado diariamente e pela disponibilidade revelada em horário laboral e extra-laboral.

A toda a Administração da Empresa Globaltronic, em especial ao Eng. Carlos Alves que permitiu que todo este projeto fosse possível e pelos bons conselhos.

Ao Dr. Paulo Bartolomeu, Diretor Técnico do Departamento de Engenharia da Empresa Globaltronic pelo grande incentivo transmitido ao longo deste trabalho e pela grande ajuda prestada na minha formação profissional.

A todos os colaboradores e colegas de trabalho da Globaltronic pelo apoio dado diariamente e pelo bom ambiente criado dentro da empresa que me permitiu crescer ainda mais como pessoa e profissional.

A todos os colegas que me acompanharam durante todo o percurso académico que me ajudaram nos momentos mais difíceis, pelas tardes e noites de estudo e todos os trabalhos efetuados em grupo.

Não menos importantes, os meus amigos mais próximos por serem os melhores, estarem sempre comigo e pela total compreensão da minha ausência em eventos festivos.

À minha namorada Catarina Marques por estar sempre ao meu lado nos momentos menos bons, por todo o carinho dado e motivação ao longo desta etapa.

Por fim, a toda a minha família, em especial aos meus pais pelo apoio incondicional que me têm dado ao longo da vida e por acreditarem sempre em mim

.

Resumo

O objetivo desta dissertação consiste no projeto de um controlador de velocidade e posição de um sistema eletromecânico utilizando realimentação de estados. Numa fase inicial, é validado o algoritmo de controlo através da simulação em MATLAB de um sistema com parâmetros físicos conhecidos. O cálculo dos ganhos de realimentação é feito recorrendo ao método de *Ackermann*, onde é utilizado o modelo em espaço de estados para representar o sistema.

Nesta dissertação, é utilizado o método recursivo dos mínimos quadrados (*RLS*) para identificação paramétrica de um sistema motor-carga, sendo posteriormente obtido o modelo discreto que representa a posição e a velocidade. O controlador por realimentação de estados foi implementado com recurso a um microcontrolador da arquitetura ARM através de um sistema de tempo real (FreeRTOS). A obtenção das variáveis de estado é efetuada por um codificador de quadratura e um sensor de corrente.

Desenvolveu-se uma interface gráfica em MATLAB, para a visualização do comportamento dinâmico do sistema e das variáveis de estado e analisar a desempenho do controlador.

Abstract

The scope of this dissertation is to design a speed and position controller of an eletromechanical system by using state feedback. Initially, the control algorithm is validated through MATLAB simulations of a system with known physical parameters. The feedback gain calculation is done by using Ackermann's method, which is the state-space model to represent the system.

In this thesis, we use the RLS (Recursive Least-squares) method for parametric identification of a motor-load system, subsequently obtained the discrete model which represents the position and velocity. The state feedback controller is implemented by using a ARM microcontroller architecture. The state variables of the system are obtained by using an encoder and a current sensor module.

It has been developed a graphical interface in MATLAB to visualize the system behaviour, the state variables and analyze the performance of the controller.

Índice

Índice	i
Lista de Acrónimos	v
Lista de Figuras	vi
Lista de Tabelas	xi
Capítulo 1 Introdução	1
1.1 Objetivos	1
1.2 Motivação.....	2
1.3 Metodologia	3
1.4 Estrutura da Dissertação	4
Capítulo 2 O Servomotor DC.....	7
2.1 Descrição de um Servomotor DC	7
2.2 Constituição de um Motor DC.....	8
2.3 Tipos de Motor DC	10
2.3.1 Motor de Excitação em Série.....	10
2.3.2 Motor de Excitação em Paralelo.....	11
2.3.3 Motor de Excitação Independente	12
2.3.4 Motor de Excitação Composta	12
2.3.5 Comparação dos motores DC	13
2.3.6 Motor de Íman permanente com escovas	13
2.3.7 Motor BLDC (<i>Brushless</i> DC Motor)	14
2.4 Sensores de Realimentação	15
2.4.1 Codificador de quadratura.....	15
2.4.2 Taquímetro	17
2.5 Malha de controlo de um Motor DC.....	17
Capítulo 3 Modelo do Sistema.....	19
3.1 Modelo elétrico do Motor DC	19
3.2 Sistema de Posicionamento	20
3.3 Representação em função de transferência no domínio contínuo	22
3.4 Representação em espaço de estados no domínio contínuo	23
3.4.1 Representação da Velocidade	24
3.4.2 Representação da Posição	25
3.5 Função de transferência do sistema no domínio discreto	26
3.6 Representação em espaço de estados no domínio discreto	27
3.6.1 Solução da Equação de Estado no domínio discreto	28

3.7 Seleção do intervalo de amostragem h	29
Capítulo 4 Simulações do Sistema	31
4.1 Descrição do método combinado	31
4.2 Parâmetros do Sistema	32
4.3 Função de Transferência do Sistema	32
4.3.1 Função de Transferência da velocidade no domínio contínuo	32
4.3.2 Função de Transferência da velocidade no domínio discreto	32
4.3.3 Função de Transferência da posição no domínio contínuo.....	33
4.3.4 Função de Transferência da posição no domínio discreto	34
4.4 Sistema com malha aberta em Espaço de Estados	34
4.4.1 Domínio Contínuo.....	34
4.4.1.1 Construção do Diagrama de Blocos em Simulink.....	35
4.4.1.2 Resposta do sistema a um degrau de referência	35
4.4.2 Domínio Discreto	36
4.4.2.1 Modelo Discreto do Sistema	36
4.4.2.2 Diagrama de Blocos em Simulink	37
4.4.2.3 Resposta do sistema discreto a um degrau de referência	37
4.4.3 Simulação Combinada	38
4.4.3.1 Diagrama de Blocos do Sistema a controlar	38
4.5 Sistema com Realimentação em Espaço de Estados	39
4.5.1 Classificação de Sistemas com Realimentação de Estados	39
4.5.2 Domínio Contínuo.....	40
4.5.2.1 Controlabilidade.....	40
4.5.2.2 Determinação dos pólos da malha fechada	41
4.5.2.3 Determinação da Matriz de Realimentação de Estados	43
4.5.2.4 Construção do Sistema Servo em Simulink.....	44
4.5.2.5 Resposta do sistema a uma posição de referência	44
4.5.3 Simulação Combinada	45
4.5.3.1 Discretização da Matriz de Realimentação de Estados.....	45
4.5.3.2 Resposta do Sistema a uma entrada de referência bipolar	46
4.5.3.3 Introdução de ruído no sistema	47
4.5.3.4 Resposta do sistema com influência de ruído	48
4.5.3.5 Introdução de uma carga externa no sistema	49
4.5.3.6 Influência de uma carga externa na resposta do sistema.....	49
4.5.3.7 Introdução de um ganho de compensação na entrada de referência.....	51
4.6 Sistema com Ação Integral.....	52
4.6.1 Domínio Contínuo.....	53
4.6.1.1 Modificação do modelo de espaço de estados.....	53
4.6.1.2 Determinação da matriz de Realimentação de estados do sistema aumentado	54
4.6.1.3 Construção do sistema servo com ação integral em Simulink.....	55
4.6.1.4 Resposta do Sistema com Ação Integral	55
4.6.2 Simulação Combinada	56

4.6.2.1	Discretização das Matrizes de Ganho	56
4.6.2.1	Obtenção do sinal de controlo do sistema servo com ação integral	57
4.6.2.2	Influência de ruído na resposta do sistema servo com ação integral.....	58
4.6.2.3	Influência de uma carga externa na resposta do sistema com ação integral..	58
4.7	Observador de Estados	60
4.7.1	Diagrama de blocos do sistema com Observador de estados.....	60
4.7.2	Estimativa do vetor de estado	61
4.7.2.1	Observabilidade	62
4.7.2.2	Método de obtenção da matriz de ganhos L	62
4.7.3	Simulação Combinada	63
4.7.3.1	Determinação dos ganhos discretos do Observador	63
4.7.3.2	Resposta do sistema a uma referência de posição	64
Capítulo 5	Plataforma de Ensaios	67
5.1	Estrutura Mecânica	67
5.2	Placa de Controlo	68
5.3	Eletrónica de atuação e de medida.....	70
5.3.1	Módulo de Potência.....	70
5.3.2	Módulo Sensor de corrente.....	70
5.3.3	Fonte de Corrente.....	73
5.4	Sistema operativo de tempo real.....	74
5.5	Software/ Firmware	74
5.5.1	Device Drivers (<i>Firmware</i>)	75
5.5.2	Tarefas	77
Capítulo 6	Identificação do Modelo Dinâmico	79
6.1	Considerações gerais sobre Identificação de Sistemas.....	79
6.2	Sinal de entrada do sistema	80
6.3	Modelos discretos do sistema.....	80
6.3.1	Classes de estruturas de modelos com representação polinomial	80
6.3.2	Obtenção do modelo de espaço de estados a partir da representação polinomial do sistema	82
6.4	Métodos de Identificação Paramétrica de Sistemas.....	82
6.4.1	Método dos Mínimos Quadrados recursivo	83
6.4.1.1	Mínimos Quadrados com esquecimento direcional	84
6.5	Resultados Experimentais	85
6.5.1	Identificação do modelo discreto da velocidade.....	85
6.5.2	Identificação do modelo discreto da posição	88
Capítulo 7	Controlo do Sistema	93
7.1	Resultados Experimentais	93

7.1.1	Controlo da Velocidade por Realimentação de Estados	93
7.1.1.1	Introdução de um torque de carga externo.....	100
7.1.2	Controlo da Posição por Realimentação de Estados	102
7.1.2.1	Introdução de um torque de carga externo.....	109
Capítulo 8	Conclusões e Trabalho Futuro	111
Bibliografia	112
Anexo A	Interface gráfica (GUI)	115
Anexo B	Fluxogramas.....	118
Anexo C	Placa de Controlo	120
Anexo D	Módulo de Potência	123

Lista de Acrónimos

ADC	Analog Digital Converter
ARM	Advanced Risc Machine
ARMAX	AutoRegressive Moving Average with eXogenous signal
ARX	AutoRegressive with eXogenous input
BLDC	Brushless Direct Current
DC	Direct Current
DIP	Dual Inline Package
GUI	Guide User Interface
LS	Least Squares
MCPWM	Motor Control Pulse Width Modulation
MOSFET	Metal Oxide Semiconductor Field Effect Transistor
MPU	Memory Protection Unit
PID	Proportional Integral Derivative
PM	Permanent Magnet
PRBS	Pseudo Random Binary Sequence
PWM	Pulse Width Modulation
QEI	Quadrature Encoder Interface
RLS	Recursive Least Squares
RTOS	Real Time Operating System
TTL	Transistor-Transistor Logic
UART	Universal Assynchronous Receiver Transmitter
ZOH	Zero Order Hold

Lista de Figuras

Figura 1 - Sistema de defesa antiaérea <i>Phalanx</i> [1]	2
Figura 2 - Diagrama de Blocos de um Servomotor DC	7
Figura 3 - Servomotor DC [6]	7
Figura 4 - Constituição básica de um motor DC [10].....	9
Figura 5 - Princípio de Funcionamento de um motor DC.....	9
Figura 6 - Representação do motor DC com excitação série	10
Figura 7 - Curva do torque-velocidade para motores de excitação série	11
Figura 8 - Representação do motor DC com excitação em paralelo.....	11
Figura 9 - Representação do motor DC com excitação independente	12
Figura 10 - Representação do motor DC com excitação composta	13
Figura 11 - Representação do motor DC de íman permanente com escovas.....	14
Figura 12 - Estrutura do motor DC de íman permanente sem escovas [5].....	14
Figura 13 - Codificador ótico rotativo [14]	15
Figura 14 - Sensor ótico Incremental [14]	16
Figura 15 - Sinal de saída do codificador incremental [15]	16
Figura 16 - Diagrama de blocos de um sistema de controlo com computador	17
Figura 17 - Modelo elétrico equivalente do motor DC	19
Figura 18 - Diagrama de um sistema de posicionamento.....	20
Figura 19 - Interligação entre o sistema a controlar e o controlador	31
Figura 20 - Resposta do Sistema em malha aberta para um degrau unitário.....	33
Figura 21 - Representação do sistema em Malha Aberta no domínio contínuo [25]	34
Figura 22 - Diagrama de blocos do sistema em malha aberta com entrada em degrau	35

Figura 23 - Resposta do Sistema contínuo em malha aberta a um degrau de referência de 12V	36
Figura 24 - Representação do sistema em malha aberta no domínio discreto [22]	36
Figura 25 - Diagrama de blocos do sistema discreto em malha aberta com entrada em degrau	37
Figura 26 - Resposta do sistema discreto em malha aberta a um sinal de referência de 12V .	38
Figura 27 - Diagrama de blocos do sistema a controlar	38
Figura 28 - Regulador de estados na forma de espaço de estados [19]	39
Figura 29 - Sistema servo na forma de espaço de estados [19]	40
Figura 30 - Diagrama do Lugar das raízes do modelo para a posição do sistema contínuo	41
Figura 31 - Resposta transitória de um sistema de segunda ordem	42
Figura 32 - Diagrama de blocos em Simulink do sistema servo sem ação integral	44
Figura 33 - Resposta do sistema servo em modo contínuo para uma referência de 40 graus.	45
Figura 34 - Resposta do sistema servo em modo combinado para uma referência de [-40,40] graus	47
Figura 35 - Diagrama de blocos do sistema a controlar com ruído Gaussiano	48
Figura 36 - Influência de ruído branco gaussiano na resposta do sistema servo	48
Figura 37 - Introdução de um torque de carga externo à planta do sistema contínuo	49
Figura 38 - Influência de um torque de carga externo de 2 N.m na resposta do sistema	50
Figura 39 - Influência de um torque de carga externo variável na resposta do sistema	50
Figura 40 - Diagrama de blocos de um sistema com realimentação total de estados	51
Figura 41 - Resposta do sistema com realimentação total de estados	51
Figura 42 - Resposta do sistema com a introdução de um ganho de pré-compensação	52
Figura 43 - Sistema Servo com Ação Integral	53
Figura 44 - Diagrama de blocos em Simulink do sistema servo com ação integral	55

Figura 45 - Resposta do Sistema Servo com Ação Integral para uma referência de 40 graus..	56
Figura 46 - Influência de ruído branco gaussiano na resposta do sistema com ação integral .	58
Figura 47 - Influência de um torque de carga fixo na resposta do sistema com ação integral	59
Figura 48 - Influência de um torque de carga variável na resposta do sistema com ação integral.....	60
Figura 49 - Diagrama de blocos do sistema servo com observador de estados	61
Figura 50 - Comportamento das variáveis estimadas pelo Observador de estados.....	64
Figura 51 - Evolução do sinal de controlo u do sistema com observador de estados	65
Figura 52 - Comportamento das variáveis de estado medidas pelo sistema	65
Figura 53 - Comportamento das variáveis de estado estimadas e medidas (intervalo t_1)	66
Figura 54 - Comportamento das variáveis de estado estimadas e medidas (intervalo t_2)	66
Figura 55 - Esquema geral da plataforma de ensaios	67
Figura 56 - Plataforma de Ensaios	68
Figura 57 - Placa de controlo do sistema real	68
Figura 58 - Módulo de Potência	70
Figura 59 - Módulo de sensor de corrente elétrica desenvolvido	71
Figura 60 - Esquema elétrico do sensor de corrente com condicionamento de sinal	71
Figura 61 - Esquema elétrico utilizado na validação do módulo de sensor de corrente	72
Figura 62 - Esquema elétrico da fonte de corrente com regulador LM317	73
Figura 63 - Processo de Identificação de Sistemas.....	79
Figura 64 - Sinal de entrada do tipo PRBS para identificação dos parâmetros do sistema	85
Figura 65 - Parâmetros identificados para obtenção do modelo discreto da velocidade	86
Figura 66 - Erro de estimação do modelo discreto da velocidade	86
Figura 67 - Sinais de saída de velocidade do sistema real e do sistema identificado	87

Figura 68 - Mapa de pólos e zeros do modelo discreto da velocidade	88
Figura 69 - Sinal de entrada do tipo PRBS para identificação do modelo discreto da posição.	89
Figura 70 - Parâmetros identificados do modelo discreto da posição	89
Figura 71 - Erro de estimação do modelo discreto da posição	89
Figura 72 - Sinais da posição do sistema real e do sistema identificado	90
Figura 73 - Mapa de pólos e zeros do modelo discreto da posição	91
Figura 74 - Sinais de saída (velocidade) e de referência do sistema real sem ação integral	94
Figura 75 - Erro do sinal de saída no controlo de velocidade do sistema real sem ação integral	95
Figura 76 - Sinal de controlo da velocidade do sistema real sem ação integral	95
Figura 77 - Sinal da variável de estado corrente do sistema real sem ação integral	96
Figura 78 - Sinais de saída (velocidade) e de referência do sistema real com ação integral	98
Figura 79 - Erro do sinal de saída no controlo de velocidade do sistema real com ação integral	98
Figura 80 - Sinal de controlo da velocidade do sistema real com ação integral	99
Figura 81 - Sinal da variável de estado corrente do sistema real com ação integral	99
Figura 82 - Influência de um torque carga externo no controlo da velocidade do sistema real	100
Figura 83 - Sinal de controlo da velocidade do sistema real com influência de uma carga ...	101
Figura 84 - Sinal da variável de estado corrente do sistema real com influência de uma carga	102
Figura 85 - Sinais de saída (posição) e de referência do sistema real com ação integral (Ensaio 1).....	104
Figura 86 - Sinais do erro de posição do sistema real com ação integral (Ensaio 1)	105
Figura 87 - Sinal de controlo da posição do sistema real com ação integral (Ensaio 1)	105
Figura 88 - Sinais de saída (posição) e de referência do sistema real com ação integral (Ensaio 2).....	107

Figura 89 - Sinais do erro de posição do sistema real com ação integral (Ensaio 2)	107
Figura 90 - Sinal de controle da posição do sistema real com ação integral (Ensaio 2)	108
Figura 91 - Influência de uma carga de 250mA no controle da posição do sistema real	110
Figura 92 - Sinal de controle da posição do sistema real com influência de um torque de carga	110

Lista de Tabelas

Tabela 1 - Comparação dos vários tipos de motores DC [13]	13
Tabela 2 - Parâmetros do Sistema.....	32
Tabela 3 - Teste de validação do módulo de sensor de corrente	72
Tabela 4 - Parâmetros estimados do modelo discreto da velocidade	86
Tabela 5 - Parâmetros estimados do modelo discreto da posição	90
Tabela 6 - Comparação dos ensaios efetuados ao sistema real no controlo de posição	108

Capítulo 1 Introdução

1.1 Objetivos

O principal objetivo da presente dissertação consiste na implementação de um controlador de velocidade e de posição com realimentação de estados, podendo ser aplicado num sistema eletromecânico real.

Numa fase inicial pretende-se desenvolver o algoritmo de controlo para um sistema com parâmetros físicos conhecidos, sendo utilizada uma ferramenta de simulação como por exemplo o MATLAB/Simulink. O controlador do sistema é implementado com base num modelo de representação por espaço de estados. Durante esta etapa de simulação, pretende-se testar a qualidade do controlador com as seguintes variantes:

- a) Controlo de posição do sistema por realimentação de estados considerando todas as variáveis de estado mensuráveis;
- b) Controlador de posição com ação integral com a introdução de ruído e variações de carga no sistema;
- c) Desenvolvimento de um observador de estados, permitindo a estimação das variáveis de estado do sistema.

O desempenho do controlador depende de requisitos definidos para a resposta transitória do sistema, como por exemplo a percentagem de sobrelevação (*overshoot*) e o tempo de estabelecimento (*settling time*). Após a etapa das simulações, pretende-se controlar um sistema real, utilizando as técnicas utilizadas na etapa de simulação. Com isto, os objetivos a cumprir na segunda etapa da dissertação são:

- 1) Utilização de uma plataforma de testes constituída por um sistema eletromecânico, eletrónica de medida e de atuação;
- 2) Desenvolvimento de um módulo para medição de corrente elétrica;
- 3) Utilização de um microcontrolador de fácil integração com sistemas de tempo real;
- 4) Identificação paramétrica de um sistema eletromecânico utilizando um método de identificação sob o sistema de tempo real. Obtenção dos modelos discretos da velocidade e posição com os parâmetros identificados;
- 5) Desenvolvimento do controlador de velocidade e posição utilizando realimentação de estados sob o sistema de tempo real;

- 6) Desenvolvimento dos controladores referidos no ponto 5) com introdução de ação integral;
- 7) Testes do sistema no controlo de velocidade e de posição com vista ao cumprimento dos requisitos definidos para a resposta transitória (tempo de estabelecimento e sobrelevação), bem como testar a validade dos pólos obtidos em malha fechada;
- 8) Testes de variação de carga do sistema real utilizando um módulo de carga.

Por fim, pretende-se visualizar os dados obtidos em tempo real com o desenvolvimento de uma interface gráfica.

1.2 Motivação

Os sistemas de controlo são utilizados em várias áreas da Engenharia e em muitas áreas da ciência. Um sistema de controlo pode ser visto de uma forma muito simples, como um interruptor de uma lâmpada ou muito complexo, como um piloto automático de um navio.

Os sistemas de controlo de posição sempre foram objeto de diversos estudos com o objetivo de se tornarem mais eficientes e precisos. Na indústria, estes sistemas de posicionamento estão presentes nas mais variadas áreas, por meio de servomotores. A área da automação industrial, da indústria têxtil e da medicina são alguns dos exemplos de intervenção no mercado dos servomotores. Porém, existe uma outra área com um crescente interesse: A área dos Sistemas de Defesa. Dentro desta área, uma aplicação de grande interesse é a Defesa Antiaérea. Um exemplo deste tipo de aplicação é o sistema de defesa antiaérea *Phalanx CIWS (Closed in Weapon System)* [1]. A figura seguinte mostra o sistema de defesa antiaérea referido anteriormente.



Figura 1 - Sistema de defesa antiaérea *Phalanx* [1]

Os servomotores são motores que seguem um determinado comando de referência, quer seja de posição ou de velocidade. Estão enquadrados na tecnologia de movimento de alto desempenho.

Um dos objetivos principais da teoria do controlo é garantir que um sistema funcione de forma adequada quando é sujeito a alterações dos parâmetros ou perturbações. No caso dos sistemas de posicionamento da Figura 1, podem ser constantemente sujeitos a perturbações externas, como por exemplo, rajadas de vento, ruídos elétricos, variações de temperatura. Com isto é necessário que um sistema seja capaz de rejeitar este tipo de perturbações de uma forma rápida e precisa.

Existem várias técnicas de como resolver os problemas descritos anteriormente. A técnica predominante na área do controlo e automação é a técnica de controlo por PID (Proporcional, integral e Derivativo). A sua utilização em larga escala advém da sua simplicidade de funcionamento associado a um alto desempenho. Além deste tipo de controlador, existem outros baseados em modelos (controladores robustos, adaptativos ou ótimos). O controlo robusto baseia-se na minimização da influência dos vários tipos de perturbações que afetam o sistema. Quanto ao controlo adaptativo, refere-se à capacidade de um sistema modificar o seu comportamento em resposta a mudanças na sua dinâmica. Quando ocorrem mudanças nos parâmetros do sistema, é realizado um auto-ajuste dos parâmetros do sistema de modo a manter o desempenho requerido.

Entre as técnicas referidas anteriormente, existem outras mais complexas como por exemplo, a técnica de controlo por lógica difusa (*Logic Fuzzy*) e por redes neuronais.

A crescente utilização dos sistemas de controlo moderno foi uma das motivações para utilizar, nesta dissertação, a representação de sistemas pelo método de espaço de estados. Este método foi desenvolvido na década de 1960 e aperfeiçoado com o aparecimento dos computadores digitais. Esta técnica permite tratar a complexidade de sistemas modernos e atender às exigências impostas pelas aplicações industriais. A utilização desta forma de representação do sistema para controlo da posição e da velocidade de um motor DC (*Direct Current*) por realimentação de estados, bem como a identificação prévia dos parâmetros de um sistema eletromecânico baseado num motor DC causaram também motivações adicionais para a realização deste trabalho.

Na implementação prática do controlo do motor DC por espaço de estados, foi utilizado um microprocessador ARM Cortex-M3. A escolha deste microprocessador foi motivada pela grande capacidade de processamento, baixo custo e baixo consumo [2].

1.3 Metodologia

Como ponto de partida nesta dissertação procedeu-se ao estudo de conceitos relacionados com motores elétricos, teoria de controlo e também de documentação produzida por outros trabalhos relacionados com a temática desta dissertação.

Após a revisão dos conceitos relacionados com o tema da dissertação, efetuaram-se as deduções das equações elétricas e mecânicas de um motor DC que permitiram obter as representações de um sistema por funções de transferência e espaço de estados. Para representação do sistema considerou-se o modelo por espaço de estados com parâmetros físicos conhecidos.

Utilizou-se como ferramenta de simulação o programa MATLAB/Simulink. Inicialmente, efetuaram-se testes ao sistema contínuo em malha aberta e de seguida procedeu-se à seleção do período de amostragem de forma a discretizar o mesmo. Posteriormente, desenvolveu-se um controlador de posição utilizando realimentação de estados com cálculo dos ganhos pelo método de *Ackermann* e de seguida introduziu-se ação integral no sistema. Efetuaram-se simulações no modo contínuo e combinado (modelo do sistema a controlar em Simulink com

um controlador discreto em MATLAB) com a possibilidade de introdução de ruído no sistema e variações de carga. Ainda na etapa de simulação desenvolveu-se um observador de estados com a estimação de todas as variáveis de estado.

Na etapa seguinte, procedeu-se à montagem da plataforma de testes, sendo esta constituída por um sistema eletromecânico (motor - carga), eletrónica de medida (codificador de quadratura e sensor de corrente) e de atuação (módulo de potência), tendo sido efetuados testes de funcionamento. Foi selecionado o microcontrolador LPC1759 [3] da arquitetura ARM, uma vez que apresenta características que se adequam aos objetivos desta dissertação. O microcontrolador selecionado possui um módulo de PWM (*Pulse Width Modulation*) dedicado ao controlo de motores (MCPWM – *Motor Control Pulse Width Modulation*) e também uma interface para um *encoder* de quadratura (QEI – *Quadrature Encoder Interface*). Sendo necessário fazer medições da corrente elétrica do sistema, desenvolveu-se um módulo de sensor de corrente e de seguida efetuaram-se testes de funcionamento.

Uma vez efetuada a montagem da eletrónica, integrou-se o sistema de tempo real FreeRTOS na placa de desenvolvimento LPC1759. Configuraram-se os *device drivers* necessários para o funcionamento do sistema. Os primeiros testes efetuados foram os de comunicação do microcontrolador com os diversos módulos (*encoder* de quadratura, sensor de corrente, módulo de potência e porta série). Estes testes foram efetuados com base numa tarefa implementada sob o FreeRTOS [4]. Desenvolveu-se em MATLAB uma interface gráfica do utilizador (GUI) para receção e visualização dos dados obtidos pelo microcontrolador. Por fim, foram implementadas as tarefas (sob o sistema de tempo real FreeRTOS) das tarefas de identificação paramétrica do sistema e, posteriormente de controlo do sistema. A etapa de implementação do controlador teve como etapa precedente a Identificação dos parâmetros do sistema eletromecânico, onde foi utilizado o algoritmo recursivo dos mínimos quadrados (RLS). Sendo o objetivo a implementação de um controlador de velocidade e de posição com realimentação de estados, foi feita a identificação dos parâmetros para obter o modelo (discreto) da velocidade e o modelo (discreto) da posição. Como referido nos objetivos, desenvolveu-se um controlador para o sistema sem ação integral e, posteriormente, introduziu-se a ação integral. Efetuaram-se testes de controlo do sistema com a visualização dos dados na aplicação GUI.

Por fim, desenvolveu-se uma fonte de corrente com o intuito de variar a carga no sistema através de um segundo motor acoplado mecanicamente ao motor DC. Com isto, efetuaram-se testes às malhas de controlo de velocidade e de posição do sistema com a introdução de carga.

1.4 Estrutura da Dissertação

A presente dissertação está organizada em oito capítulos: introdução, o motor DC, definição do modelo do sistema, simulações do sistema, plataforma de ensaios, identificação do sistema, controlo do sistema e, por fim, conclusões e trabalho futuro.

No capítulo introdutório é efetuada uma descrição dos objetivos do trabalho, motivação e metodologia utilizada neste projeto.

No segundo capítulo é efetuada uma abordagem ao motor elétrico de corrente contínua (DC), bem como o seu princípio de funcionamento. São descritos os vários tipos de motor DC e, por fim, apresentam-se os servomotores DC, descrevendo os seus elementos constituintes e as vantagens/desvantagens de utilização.

O terceiro capítulo subdivide-se em várias partes: primeiramente apresenta-se o modelo elétrico do sistema servo utilizado em ambiente de simulação, de seguida deduzem-se as equações dinâmicas do sistema e também as representações do sistema por espaço de estados e função de transferência. Por fim, é descrito o critério de seleção do período de amostragem a ser aplicado tanto na fase das simulações como na fase da implementação prática.

No quarto capítulo são apresentados os parâmetros físicos do sistema a simular em MATLAB/Simulink. É descrita detalhadamente a metodologia de cálculo dos ganhos de realimentação de estados, bem como o processo de discretização do modelo em espaço de estados do sistema de modo a efetuarem-se as simulações em modo combinado. Também é descrito o processo de modelação do sistema com ação integral procedendo de seguida ao cálculo dos ganhos de realimentação de estados. Apresentam-se os resultados obtidos das simulações efetuadas no controlo da posição do sistema servo em várias vertentes: sem ação integral, com ação integral, possibilidade de introduzir ruído e carga. No final do capítulo, descreve-se detalhadamente o método de desenvolvimento de um observador de estados e por fim, apresentam-se os resultados de simulação obtidos com a sua utilização.

No quinto capítulo descreve-se a plataforma de ensaios, começando por apresentar a sua estrutura eletromecânica e de seguida toda a eletrónica utilizada e desenvolvida para o funcionamento correto do sistema, tendo em conta os objetivos pretendidos para esta dissertação. Numa fase posterior é feita uma abordagem aos principais conceitos referentes aos sistemas operativos de tempo real, dando um maior ênfase ao FreeRTOS. Neste capítulo apresenta-se o *firmware/software* implementado ao nível das configurações dos *device drivers*, identificação e controlo do sistema.

No sexto capítulo apresentam-se fundamentos teóricos relacionados com identificação de sistemas, seguindo-se da apresentação dos resultados obtidos do processo de identificação paramétrica do modelo do sistema real. Posteriormente apresentam-se os modelos discretos (da velocidade e da posição) obtidos do processo referido anteriormente.

O sétimo capítulo descreve a implementação dos controladores de velocidade e de posição da plataforma utilizada. De seguida, são descritos os resultados dos ensaios efetuados ao sistema real com a implementação dos controladores discretos. Neste capítulo também é efetuada uma análise ao desempenho dos controladores e do sistema real.

No oitavo capítulo são apresentadas as conclusões referentes ao trabalho realizado, bem como propostas de trabalho futuro.

Capítulo 2 O Servomotor DC

No presente capítulo é feita uma descrição detalhada dos servomotores DC e dos seus elementos constituintes. Relativamente à sua constituição, é efetuada a descrição de um motor DC, do princípio de funcionamento e dos vários tipos de motores DC. De seguida, apresentam-se os dispositivos utilizados na malha de controlo e, por fim, descreve-se esta malha de controlo, que se pretende digital.

2.1 Descrição de um Servomotor DC

Nesta seção são abordados os servomotores. Existem diversos tipos de servomotores desde servomotores DC (com escovas e sem escovas), servomotores AC, servomotores de passo, servomotores de indução [5].

O Servomotor DC é basicamente um motor DC (atuador) com a adição de dispositivos tais como sensores de velocidade ou posição, redutor de velocidade e eletrónica de controlo e potência. Este tipo de motor pode ser aplicado nas mais diversas áreas destacando-se os sistemas de posicionamento, robótica e máquinas CNC. A dimensão deste tipo de motores depende do tipo de aplicação e a precisão depende da eletrónica a utilizar (sensores de medição de posição e/ou velocidade).

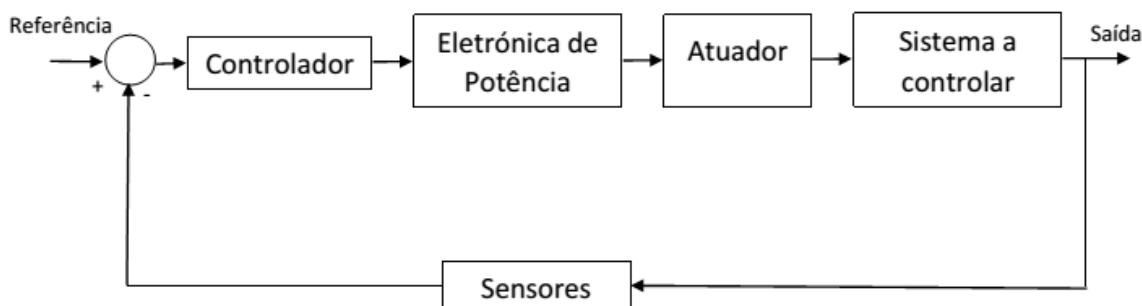


Figura 2 - Diagrama de Blocos de um Servomotor DC



Figura 3 - Servomotor DC [6]

Na Figura 2 encontra-se representado o diagrama de blocos de um Servomotor DC. Este é constituído essencialmente por um controlador (do tipo digital), eletrónica de potência (ou de atuação), atuador (motor DC), sistema a controlar e sensores.

A Figura 3 mostra a constituição de um servomotor DC com sensores de medição num dos extremos do eixo para obtenção da posição e/ou velocidade (codificador de quadratura e taquímetro). Existem outros elementos importantes que constituem um servomotor DC e que não estão representados na figura anterior, como por exemplo uma fonte de alimentação DC, um redutor de velocidade e um módulo de potência constituído essencialmente por um circuito do tipo *chopper* (ou ponte H). No módulo de potência (ou de atuação), o *chopper* permite regular o fluxo de corrente que entra no motor DC. Para aplicações em que se pretende efetuar acelerações e travagens do motor DC, é utilizado um *chopper* de quatro quadrantes. Para mais detalhes sobre o seu funcionamento, consultar em [7].

Este tipo de servomotores apresenta vantagens de utilização. No caso dos servomotores DC com escovas, estes funcionam tendo em conta uma vasta gama de velocidades e o seu controlo é relativamente simples. Não há consumo de energia quando o motor está parado e não tem carga aplicada [8]. O custo associado ao módulo de potência do motor é relativamente baixo, necessitando apenas de uma ponte H para o controlo de potência do motor. Para o caso dos servomotores DC sem escovas, estes não necessitam de manutenção visto que não têm escovas e comutador. Por fim, os servomotores sem escovas podem ser utilizados em aplicações com elevada dinâmica, não necessitando de uma eletrónica de potência muito complexa.

Relativamente às desvantagens dos servomotores DC, no caso de serem utilizadas escovas, existe um grande desgaste destas, principalmente se o motor funcionar em sobrecarga [9]. Outra desvantagem é a possível desmagnetização do motor com o excesso de corrente elétrica. Uma vez que o motor DC funciona com uma malha de realimentação, é necessária a utilização de sensores de posição e/ou velocidade, aumentando assim o custo da solução.

2.2 Constituição de um Motor DC

Um motor DC (*Direct Current*) é uma máquina elétrica com a capacidade de converter energia elétrica em energia mecânica através da interação de um campo magnético com condutores de corrente elétrica.

Os componentes básicos de um motor DC são apresentados na figura seguinte. Estes motores são constituídos pelo rotor, o estator, o comutador (coletor) e as escovas. De salientar que o rotor inclui um comutador que está em contato elétrico com as escovas.

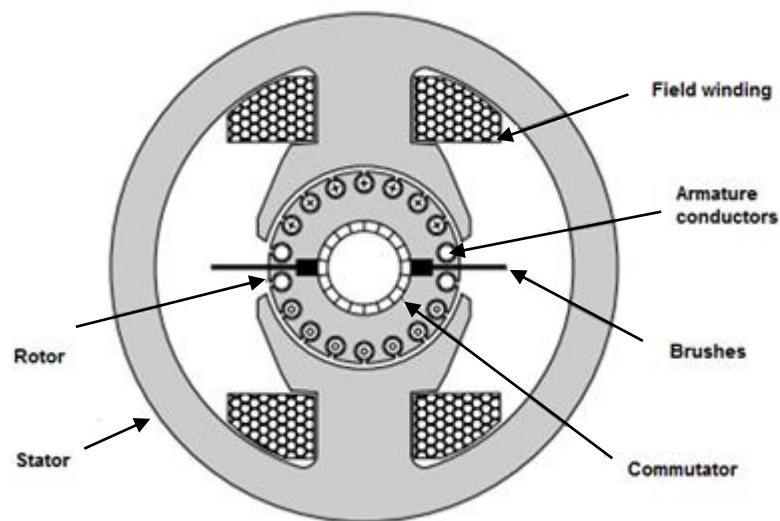


Figura 4 - Constituição básica de um motor DC [10]

O Estator (*stator*) contém um enrolamento, chamada bobina de campo (*field winding*), que é alimentado diretamente por uma fonte de tensão contínua. No caso de pequenos motores, o estator pode ser um simples ímã permanente. O *Rotor* é constituído por um enrolamento, chamado armadura (*armature conductors*), que é alimentado por uma fonte de tensão contínua através do coletor ou comutador (*commutator*) e de escovas de grafite (*brushes*). O comutador (coletor) é um dispositivo elétrico ao qual estão conectados os terminais das espiras da armadura. A figura apresentada de seguida mostra o funcionamento de um motor DC [11].

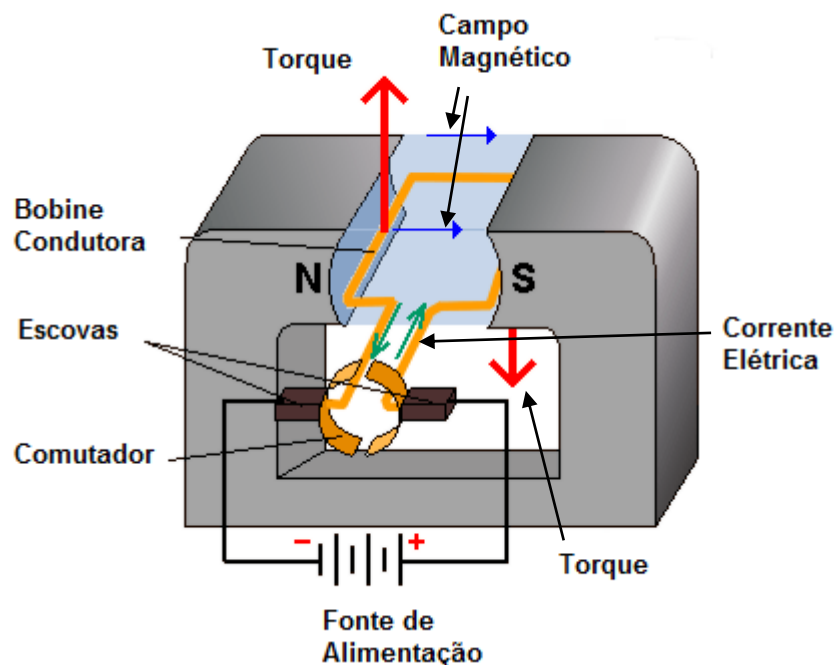


Figura 5 - Princípio de Funcionamento de um motor DC

Todos os motores elétricos são baseados nos princípios do eletromagnetismo. Num motor existem dois eletroímãs, em que um impulsiona o outro. Um dos eletroímãs de um motor tem uma posição fixa, isto é, encontra-se ligado à armação externa do motor e é denominado de campo. O outro eletroímã tem o nome de armadura e encontra-se no eixo de rotação do motor.

Quando o motor é ligado, a corrente elétrica percorre a bobina do campo magnético, determinando assim os pólos norte (N) e sul (S) da Figura 5. É também fornecida corrente elétrica ao ímã da armadura. O ímã da armadura, tendo um movimento livre, gira até que o seu pólo norte se aproxime do pólo sul do ímã do campo. Se nada mais acontecesse, o motor pararia por completo. A corrente é invertida no eletroímã da armadura (com o uso de um comutador) um pouco antes de os pólos opostos se encontrarem. Com a inversão da corrente no eletroímã da armadura, é também invertida a posição dos seus pólos, ou seja, o pólo norte passa a ser o que está mais próximo do pólo norte da armadura e estes repelem-se permitindo que o motor continue em movimento.

2.3 Tipos de Motor DC

2.3.1 Motor de Excitação em Série

Nesta configuração do motor de corrente contínua, a bobina de campo encontra-se em série com a armadura. O enrolamento de campo é caracterizado por ter um número reduzido de espiras mas de seção elevada de forma a permitir elevada intensidade de corrente elétrica que nelas circula. A figura seguinte mostra a representação esquemática do motor de excitação série.

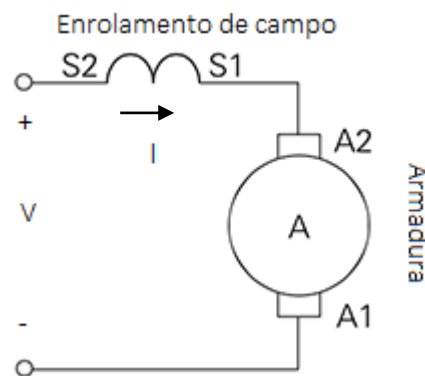


Figura 6 - Representação do motor DC com excitação série

Este tipo de motores caracteriza-se por apresentar um torque inicial elevado sendo, por isso, utilizado para arranques de cargas elevadas. A figura seguinte mostra a curva característica do torque em função da velocidade para os motores de excitação série [10].

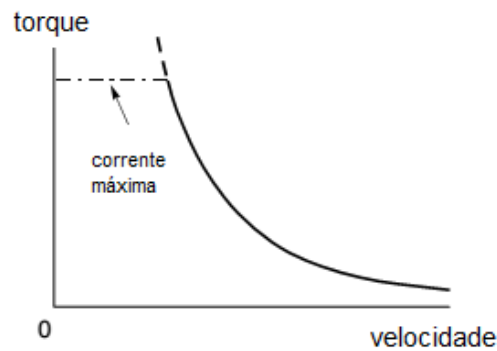


Figura 7 - Curva do torque-velocidade para motores de excitação série

A velocidade do motor é inversamente proporcional à intensidade de corrente elétrica na armadura. O torque é proporcional ao quadrado da corrente elétrica da armadura uma vez que a corrente que atravessa a bobina de campo é igual à corrente que atravessa a armadura [7].

De referir ainda que neste tipo de motores para valores de cargas baixas a velocidade do motor poderá aumentar para além de limites estáveis, podendo assim danificar o motor.

2.3.2 Motor de Excitação em Paralelo

Nos motores de excitação em paralelo, as bobinas de campo encontram-se ligadas em paralelo com a armadura, como se pode verificar na figura seguinte.

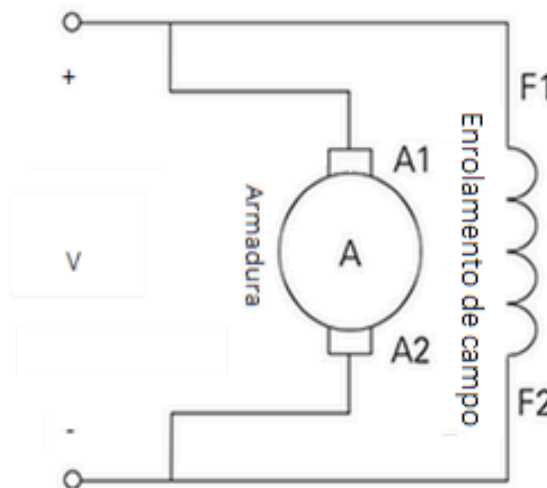


Figura 8 - Representação do motor DC com excitação em paralelo

Nesta configuração, é apenas necessária uma fonte de alimentação dado que a armadura do motor e o enrolamento de campo encontram-se em paralelo. Este tipo de motor é um motor de velocidade facilmente controlável mesmo com variações de carga. Este tipo de motor não é adequado para aplicações de velocidade variável [7] pela razão apresentada anteriormente.

O aumento do torque provoca um aumento da corrente elétrica na armadura e, consequentemente, a queda de tensão na armadura diminui ao mesmo tempo que a força contra eletromotriz diminui [7].

2.3.3 Motor de Excitação Independente

A corrente do enrolamento de campo é fornecida a partir de uma fonte de alimentação diferente da utilizada para a alimentação da armadura do motor. Na figura seguinte, encontra-se a representação desta configuração do motor.

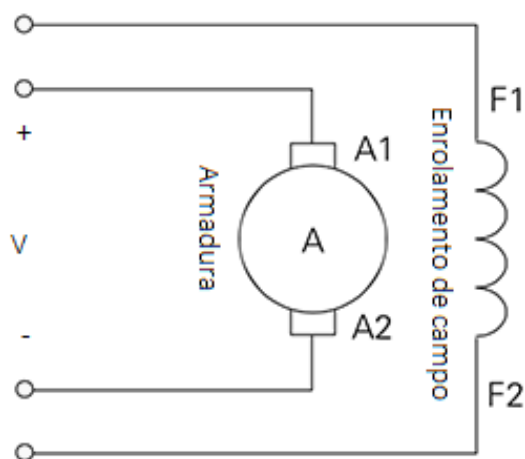


Figura 9 - Representação do motor DC com excitação independente

Uma vez que a tensão induzida na armadura é independente do fluxo no enrolamento de campo, é possível controlar estes dois parâmetros de forma independente. Com esta configuração, é possível controlar de uma forma mais fácil a velocidade do motor.

2.3.4 Motor de Excitação Composta

Relativamente ao motor de excitação composta, o enrolamento de campo encontra-se dividido, isto é, uma parte é ligada em série com a armadura do motor e a outra em paralelo. A representação deste tipo de motor é apresentada na figura seguinte.

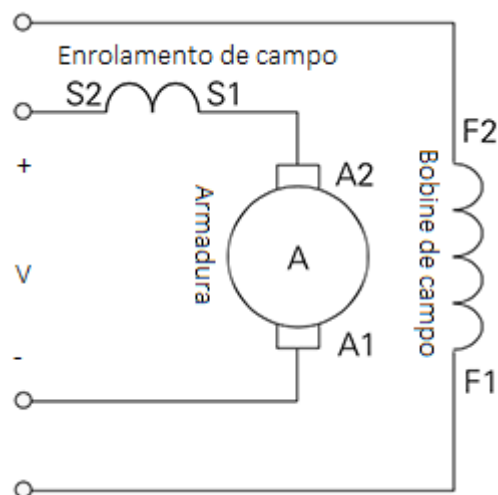


Figura 10 - Representação do motor DC com excitação composta

Este tipo de motor combina características do motor de excitação série e do motor de excitação paralela, podendo ter duas configurações, relativamente ao fluxo do enrolamento do campo: Diferencial e Cumulativo [12].

2.3.5 Comparação dos motores DC

A seguinte tabela mostra de forma resumida as diferentes características dos motores abordados nas seções anteriores.

Tipo de Motor	Torque de Arranque	Velocidade	Aplicações
Série	Elevado	Variável	Tração mecânica e elevadores
Paralelo	Baixo	Constante	Máquinas-ferramentas
Excitação independente	Baixo	Variável	Conjuntos Ward-Leonard, Sistemas atuadores especiais
Excitação Composta	Elevado	Pouco variável	Elevadores

Tabela 1 - Comparação dos vários tipos de motores DC [13]

2.3.6 Motor de Ímã permanente com escovas

O motor de ímã permanente (PM – *Permanent Magnet*) é conhecido por ter o fluxo magnético constante qualquer que seja o valor da corrente elétrica na armadura do motor. O campo magnético é gerado por um ímã permanente, apresentando assim características de velocidade, de torque e de corrente lineares. Este tipo de motor é frequentemente usado em aplicações de baixa potência.

Este tipo de motores apresentam uma vida útil reduzida devido ao atrito que se verifica no conjunto escova-comutador, podendo causar interferências eletromagnéticas nos circuitos

eletrônicos. O desgaste inerente à utilização das escovas é outra desvantagem deste motor [12].

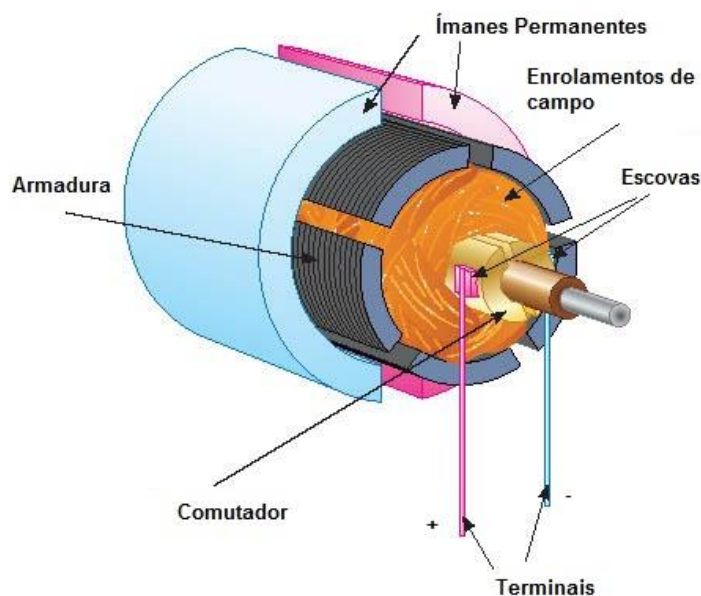


Figura 11 - Representação do motor DC de ímã permanente com escovas

2.3.7 Motor BLDC (*Brushless* DC Motor)

A abordagem feita na seção 2.2.6 foi referente a um motor de ímã permanente com escovas. Quando o motor não tem escovas, designa-se de motor *brushless*. A ausência de escovas neste motor é vantajosa na medida em que a probabilidade deste sofrer um curto-circuito é nula. Neste tipo de motor a operação de comutação é feita por um comutador eletrônico (transistor bipolar de porta isolada, transistor de efeito de campo), ao contrário do que acontece no motor DC de ímã permanente com escovas.

A figura seguinte mostra a estrutura do motor de ímã permanente sem escovas.

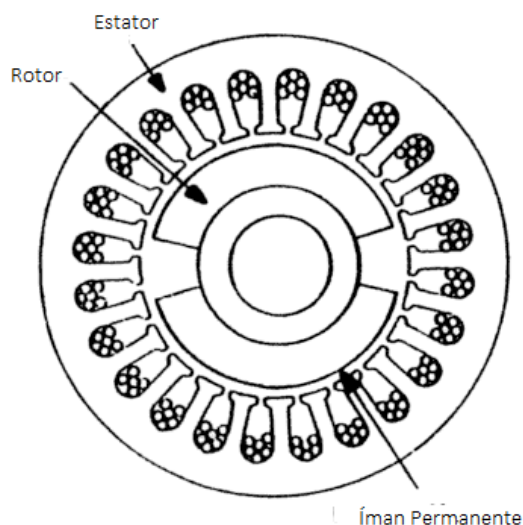


Figura 12 - Estrutura do motor DC de ímã permanente sem escovas [5]

Este tipo de motor apresenta duas configurações no que se refere à localização do íman no rotor. Isto é, o íman pode estar localizado na superfície do rotor ou então pode estar embutido no rotor. Esta segunda configuração é mais robusta que a primeira.

O motor BLDC apresenta uma baixa manutenção, um bom rendimento, uma vida útil longa e tem um desgaste mecânico muito baixo pelo fato de não possuir escovas. A inércia deste motor é bastante baixa pelo fato de ser constituído por um rotor mais leve. Isto permite um melhoramento na aceleração e travagem. Como desvantagem, o custo de projeto aumenta devido ao fato de ser necessário um comutador eletrónico. Relativamente às aplicações dos motores *brushless*, estes encontram-se por exemplo nas máquinas de lavar, nos compressores, nos leitores de CD/DVD existentes no computador.

2.4 Sensores de Realimentação

Nesta seção faz-se uma breve descrição dos dispositivos que tornam o servomotor DC do tipo *closed-loop*, isto é, realimentado. Estes dispositivos são denominados por sensores de realimentação. São eles: o codificador de quadratura (*encoder*), o taquímetro, entre outros.

2.4.1 Codificador de quadratura

O codificador de quadratura (ou *encoder*) é um sensor que converte um movimento linear ou angular em sinais digitais. Esses sinais digitais podem ser utilizados para determinar a posição, a direção, a velocidade e a aceleração do sistema ao qual o codificador está acoplado.

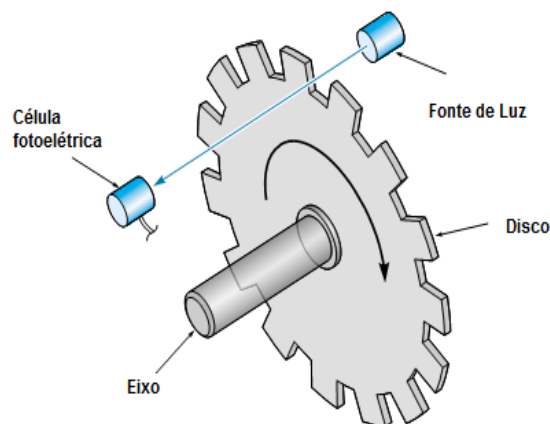


Figura 13 - Codificador ótico rotativo [14]

Os codificadores óticos apresentam muitas aplicações: as mais comuns são nos eixos dos robôs, no posicionamento de antenas, radares e no controle de velocidade e posição de motores elétricos. O codificador ótico, representado na Figura 13, é classificado em dois tipos: incremental ou absoluto.

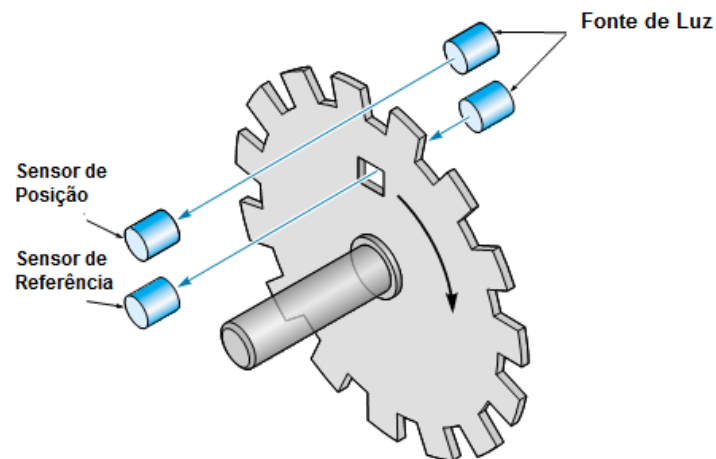


Figura 14 - Sensor óptico Incremental [14]

O sensor óptico incremental possui um conjunto de “espaços vazios” que se encontram igualmente espaçados, como se mostra na Figura 14. Cada “espaço vazio” representa um ângulo [14]. O sensor óptico incremental, também conhecido como sensor em quadratura possui dois canais em forma de onda quadrada e a sua saída fornece informações sobre o movimento executado e a direção de rotação do motor. Este tipo de codificador gera um certo número de impulsos por rotação. Os dois sinais de onda quadrada têm o nome de canal A e B e estes são gerados com um desfasamento de 90° . A leitura do terceiro sinal, denominado de Z ou zero, dá informação sobre a posição específica, ou seja, a posição zero. Este sinal é um pulso que é acionado cada vez que o sensor passa pela posição de referência. Para determinação da posição é necessário haver um controlo eletrónico do sinal dado que o codificador óptico incremental gera somente uma sequência de impulsos. De referir ainda que este sinal é um pulso de onda quadrada gerado a cada 360° . A figura seguinte mostra as ondas do canal A e B do sensor incremental. Pode-se observar o desfasamento de 90° entre o canal A e B, conforme o sentido de rotação.

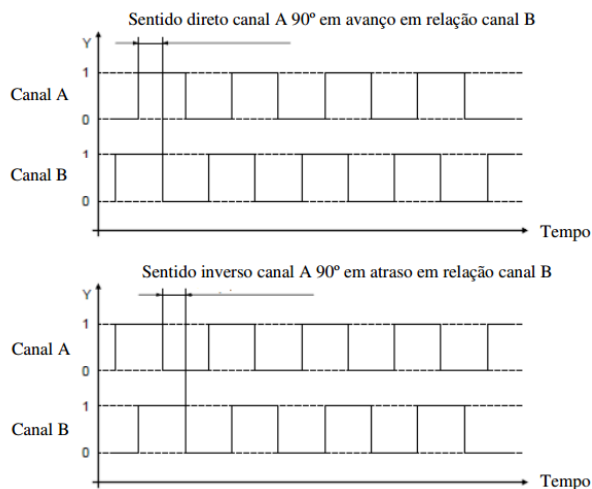


Figura 15 - Sinal de saída do codificador incremental [15]

O codificador ótico absoluto, tal como o incremental, é usado na determinação da posição. Ao contrário do codificador ótico incremental, que fornece impulsos desde uma posição de referência, o codificador absoluto fornece um código binário que representa a posição [14]. Uma vantagem de utilização deste tipo de codificador é que a posição real não é perdida no caso de perda de alimentação. Caso haja perda de alimentação, não é necessário voltar à posição zero para determinar a posição, como acontece no codificador incremental. Este tipo de codificador requer a montagem de células fotoelétricas e que estas sejam alinhadas com precisão [14].

2.4.2 Taquímetro

O taquímetro é um transdutor que tem como função converter o sinal de rotação do motor num sinal de tensão, isto é, um gerador DC. Este tipo de taquímetro é designado de taquímetro DC. Tipicamente, estes dispositivos apresentam ímanes permanentes. O taquímetro é acoplado ao eixo do motor DC que funciona como um sensor de velocidade fornecendo ao controlador eletrónico o valor de tensão proporcional à velocidade do motor. O acoplamento deste tipo de taquímetros aos motores DC é feito de forma direta (*piggyback*), uma vez que o encapsulamento dos taquímetros assim o permite [14].

2.5 Malha de controlo de um Motor DC

A implementação de um controlador com realimentação é efetuada num computador digital. Uma vez que o computador recebe dados apenas em intervalos de tempo específicos, é necessário desenvolver um método que descreva e analise o desempenho dos sistemas de controlo por computador [16]. A figura seguinte mostra o diagrama de blocos de um sistema de controlo digital geral com apenas uma malha.

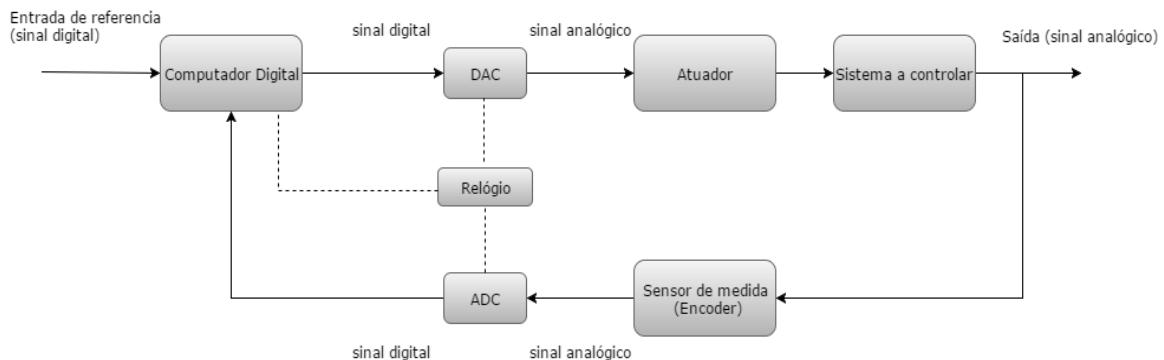


Figura 16 - Diagrama de blocos de um sistema de controlo com computador

Na configuração do sistema apresentada na Figura 16, o elemento atuador é um motor DC. O computador digital recebe como entrada um sinal digital e executa cálculos de modo a fornecer uma saída em forma digital. O computador pode ser programado para fornecer uma saída em que o desempenho do processo a controlar seja próximo do desempenho desejado. Um computador digital recebe e faz o tratamento dos sinais em forma digital (numérica), em contraste com os sinais contínuos. Um sistema de controlo digital usa sinais digitais e um computador digital para fazer o controlo de processos. Os dados de medição são convertidos

da forma analógica para a forma digital através de um conversor analógico-digital (ADC). Depois de processar os sinais de entrada, o computador digital fornece uma saída em forma digital. Esta saída é convertida na forma analógica, através de um conversor digital-analógico (DAC). Normalmente, nestes sistemas existe um sinal de relógio, que pode ser interno ou externo e que é utilizado para amostrar a saída do sistema com um período de amostragem h .

O controlo de sistemas físicos por computador oferece vantagens significativas relativamente a sistemas analógicos, tais como o aumento de desempenho, redução de custos, fiabilidade e flexibilidade. A possibilidade de aumentar a complexidade dos controladores permite aumentar o desempenho do sistema de controlo. Dada a evolução industrial que se tem feito sentir, é possível adquirir processadores digitais a custo reduzido. Por fim, a capacidade de alterar o controlador por *software*, reflete-se numa estratégia flexível de ajuste dos sistemas de controlo.

Capítulo 3 Modelo do Sistema

O modelo dinâmico do motor é indispensável para a análise e projeto de um controlador a ser implementado, neste caso um controlador de velocidade ou posição por realimentação de estados. Este modelo obtém-se a partir das características elétricas e mecânicas do motor DC.

No presente capítulo são apresentadas as equações do sistema dinâmico formado por um motor DC e um sistema de engrenagens e, por fim, é apresentado o modelo deste sistema na forma de espaço de estados.

3.1 Modelo elétrico do Motor DC

Para simular um sistema físico, é necessário obter um modelo que represente a sua dinâmica. No caso do motor elétrico DC, esse modelo é descrito sob a forma de uma equação diferencial de segunda ordem que é obtida pela conjugação das leis de Newton para a rotação e das leis de *Kirchoff* para os circuitos elétricos. A figura seguinte mostra o modelo elétrico do motor DC na sua forma equivalente. No lado direito da figura encontra-se representada a componente mecânica do motor e no lado esquerdo a componente elétrica do motor.

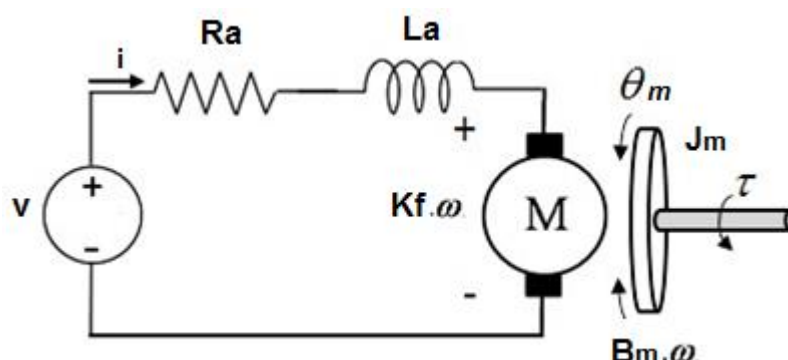


Figura 17 - Modelo elétrico equivalente do motor DC

Quando o motor elétrico começa a rodar, surge uma força contra-eletromotriz induzida nas espiras das bobinas que constituem a armadura. Isto deve-se ao fenómeno de indução eletromagnética, que acontece quando o campo magnético perto de um condutor elétrico varia, provocando o aparecimento de uma tensão nos terminais desse condutor. A força contra-eletromotriz induzida na armadura é proporcional à velocidade ω do motor. De acordo com a Figura 17, a força contra eletromotriz v_{em} é dada por:

$$v_{em}(t) = K_f \omega(t) \quad (3.1)$$

Na equação anterior, a constante de proporcionalidade é a constante de força contra eletromotriz K_f , expressa em V/(rad/s) e $\omega(t)$ é a velocidade do rotor em rad/s.

Relativamente à componente mecânica, o rotor gira no interior do motor e possui uma determinada massa e geometria. O rotor tem também um dado momento de inércia, que é

dado por J_m . Ainda na componente mecânica, entre a parte fixa e a parte móvel do motor, existe uma componente chamada atrito. Esta componente de atrito pode ser denominada de coeficiente de atrito viscoso dos rolamentos do motor e é dada por B_m .

Na componente elétrica, a armadura do motor é modelada tendo em conta uma resistência constante R_a em série com uma indutância L_a , que representa a indutância do enrolamento da armadura. A tensão de alimentação do motor é dada por v . Por fim, a corrente que circula no motor é representada por i e o torque desenvolvido pela armadura do motor por τ .

Pela lei das malhas de *Kirchhoff* obtém-se:

$$v(t) = R_a i(t) + L_a \frac{di(t)}{dt} + K_f \omega(t) \quad (3.2)$$

Por sua vez, o torque desenvolvido pela armadura do motor é dado por [18]:

$$\tau(t) = K_m i(t) = J_m \frac{d\omega(t)}{dt} + B_m \omega(t) + \tau_L(t) \quad (3.3)$$

onde K_m é a constante de torque desenvolvido pelo motor expressa em Nm/A e τ_L , expresso em Nm, é o torque de carga que é aplicado externamente ao motor.

3.2 Sistema de Posicionamento

A figura seguinte [17] representa o diagrama de blocos de um sistema de posicionamento, sendo este o sistema que se pretende controlar.

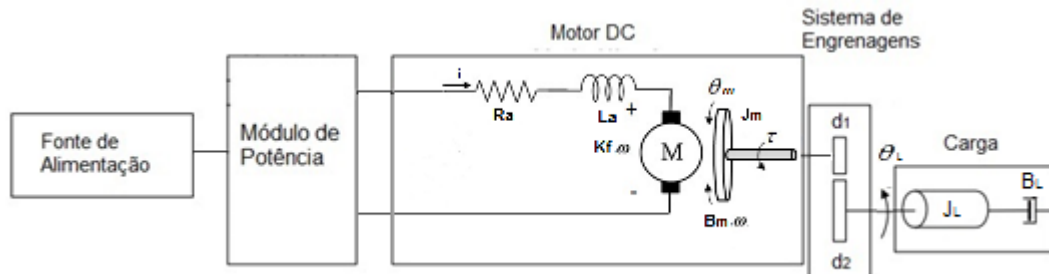


Figura 18 - Diagrama de um sistema de posicionamento

O sistema de posicionamento apresentado é constituído por uma caixa de redução de velocidade que é representado por um sistema de engrenagens. Na figura apresentada não se encontra a componente eletrónica de realimentação (sensores de medição), sendo esta essencial na implementação de um controlador. A caixa de redução de velocidade, na sua forma simples apresenta duas rodas dentadas, sendo que a roda dentada do eixo do motor tem um diâmetro menor que a roda dentada do eixo da carga. Isto permite reduzir a velocidade aumentando por sua vez o torque. As variáveis do momento de inércia e do coeficiente de atrito da carga são representados por J_L e B_L , respetivamente.

A relação entre o torque transmitido à engrenagem e_2 e o torque transmitido à engrenagem e_1 é dada por [18]:

$$\frac{\tau_{e2}(t)}{\tau_{e1}(t)} = \frac{N_2}{N_1} \quad (3.4)$$

Da equação 3.4, tem-se que

$$\tau_{e2}(t) = \tau_{e1}(t) \frac{N_2}{N_1} \quad (3.5)$$

A equação anterior representa o torque desenvolvido à saída do sistema de engrenagens, que depende do torque produzido pelo motor e da relação das duas engrenagens $\frac{N_2}{N_1}$. A relação de torque entre as duas engrenagens também pode ser dada em função do deslocamento angular, como se pode verificar de seguida.

$$\frac{\tau_{e2}(t)}{\tau_{e1}(t)} = \frac{\Theta_m}{\Theta_L} \quad (3.6)$$

A combinação das equações 3.4 e 3.6 é dada pela seguinte expressão, que relaciona o deslocamento angular do eixo do motor e da carga com o número de dentes de cada roda dentada.

$$\frac{\Theta_m}{\Theta_L} = \frac{N_2}{N_1} \quad (3.7)$$

Uma vez que o sistema da Figura 18 é constituído por um mecanismo de acoplamento do motor DC à carga através de um sistema de engrenagens, o torque produzido pela armadura do motor é dado pela seguinte equação:

$$\tau(t) = K_m i(t) = J_m \frac{d\omega(t)}{dt} + B_m \omega(t) + \tau_{e1}(t) \quad (3.8)$$

Na equação 3.8, pode-se representar $\left(\frac{d\omega(t)}{dt}\right)$ e $(\omega(t))$ por $\ddot{\Theta}_m$ e $\dot{\Theta}_m$, respetivamente. Substituindo a equação 3.5 na equação 3.8,

$$\tau(t) = J_m \ddot{\Theta}_m + B_m \dot{\Theta}_m + \tau_{e2}(t) \frac{N_1}{N_2} \quad (3.9)$$

O torque transmitido à engrenagem e_2 , dependente do torque de carga τ_L é dado pela equação:

$$\tau_{e2}(t) = J_L \ddot{\Theta}_L + B_L \dot{\Theta}_L + \tau_L(t) \quad (3.10)$$

Substituindo a equação 3.10 na equação 3.9,

$$\tau(t) = J_m \ddot{\Theta}_m + B_m \dot{\Theta}_m + (J_L \ddot{\Theta}_L + B_L \dot{\Theta}_L + \tau_L(t)) \frac{N_1}{N_2} \quad (3.11)$$

De acordo com a expressão 3.7, a equação anterior pode ser escrita em função de $\ddot{\theta}_m$ e $\dot{\theta}_m$, tem-se,

$$\tau(t) = \left(J_m + J_L \left(\frac{N_1}{N_2} \right)^2 \right) \ddot{\theta}_m + \left(B_m + B_L \left(\frac{N_1}{N_2} \right)^2 \right) \dot{\theta}_m + \tau_L(t) \left(\frac{N_1}{N_2} \right) \quad (3.12)$$

Por fim, apresenta-se a equação 3.12 na forma geral [19].

$$\tau(t) = J_o \frac{d\omega(t)}{dt} + B_o \omega(t) + \tau_L(t) \left(\frac{N_1}{N_2} \right) \quad (3.13)$$

Na equação anterior, J_o representa o momento de inércia da combinação do motor DC, sistema de engrenagens e carga e B_o representa o coeficiente de atrito resultante desta combinação.

3.3 Representação em função de transferência no domínio contínuo

Para representar um sistema na forma de função de transferência, será necessário aplicar as Transformadas de Laplace. Sendo assim, aplicando as Transformadas de Laplace às equações 3.2 e 3.13,

$$V(s) = R_a I(s) + L_a I(s) s + K_f W(s) \quad (3.14)$$

$$\tau(s) = K_m I(s) = J_o \frac{dW(s)}{dt} + B_o W(s) + \tau_L(s) \left(\frac{N_1}{N_2} \right) \quad (3.15)$$

Considera-se que a relação entre as duas engrenagens é dada por $n = \left(\frac{N_1}{N_2} \right)$. Resolvendo a equação 3.15 em ordem a $I(s)$,

$$I(s) = \frac{J_o W(s)s + B_o W(s) + \tau_L(s)n}{K_m} \quad (3.16)$$

Substituindo a equação 3.16 na equação 3.14,

$$V(s) = (R_a + L_a s) \frac{J_o W(s)s + B_o W(s) + \tau_L(s)n}{K_m} + K_f W(s) \quad (3.17)$$

A equação 3.17 na sua forma simplificada é da forma,

$$V(s)K_m - (R_a + L_a s)\tau_L(s)n = W(s)[(R_a + L_a s)(J_o s + B_o) + K_m K_f] \quad (3.18)$$

Resolvendo a equação 3.18 em ordem a $w(s)$,

$$W(s) = \frac{V(s)K_m}{[(R_a + L_a s)(J_o s + B_o) + K_m K_f]} - \frac{(R_a + L_a s)\tau_L(s)n}{[(R_a + L_a s)(J_o s + B_o) + K_m K_f]} \quad (3.19)$$

Considerando que o sistema tem $\tau_L = 0$, isto é, não tem carga aplicada, a equação 3.19 pode ser representada da seguinte forma:

$$W(s) = G_{11}(s) V(s) \quad (3.20)$$

A função de transferência da velocidade à saída do sistema de engrenagens considerando a relação entre as duas engrenagens, dada por n .

$$\frac{W(s)}{V(s)} = G_{11}(s) n = \frac{K_m n}{[(R_a + L_a s) (J_o s + B_o) + K_m K_f]} \quad (3.21)$$

A função de transferência da velocidade corresponde a um sistema de segunda ordem. Multiplicando a função $G_{11}(s)$ por $\left(\frac{1}{s}\right)$ no domínio de Laplace, obtém-se a função de transferência para a posição, dado que a posição é o integral da velocidade.

$$\frac{\Theta(s)}{V(s)} = G_{21}(s) = \frac{K_m n}{[(R_a + L_a s) (J_o s + B_o) + K_m K_f]} \left(\frac{1}{s}\right) \quad (3.22)$$

Sendo assim, para a posição obtém-se um sistema de terceira ordem.

3.4 Representação em espaço de estados no domínio contínuo

Nesta seção é descrito o processo de obtenção da representação em espaço de estados do sistema. Como já referido anteriormente, um dos objetivos da presente dissertação é a análise e simulação da dinâmica do sistema pelo método de espaço de estados. As equações 3.2 e 3.13 são reescritas na seguinte forma:

$$\frac{di(t)}{dt} = \frac{v(t)}{L_a} - \frac{R_a}{L_a} i(t) - \frac{K_f}{L_a} \omega(t) \quad (3.23)$$

$$\frac{d\omega(t)}{dt} = -\frac{B_o}{J_o} \omega(t) + \frac{K_m}{J_o} i(t) - \frac{\tau_L(t) n}{J_o} \quad (3.24)$$

Colocando as equações 3.23 e 3.24 em ordem à variável torque τ , tem-se

$$\frac{d\tau(t)}{dt} = \frac{K_m}{L_a} V(t) - \frac{R_a}{L_a} \tau(t) - \frac{K_m K_f}{L_a} \omega(t) \quad (3.25)$$

$$\frac{d\omega(t)}{dt} = -\frac{B_o}{J_o} \omega(t) + \frac{1}{J_o} \tau(t) - \frac{\tau_L(t) n}{J_o} \quad (3.26)$$

As equações dinâmicas de estado referentes ao método de espaço de estados são as seguintes:

$$\dot{\mathbf{x}} = \mathbf{A} \mathbf{x} + \mathbf{B} u \quad (3.27)$$

$$y = \mathbf{C} \mathbf{x} + \mathbf{D} u \quad (3.28)$$

Das equações 3.27 e 3.28 apresentadas anteriormente, \mathbf{A} representa a matriz de estado, \mathbf{B} a matriz de entrada, \mathbf{C} a matriz de saída e \mathbf{D} a matriz de avanço (*feedforward*). O vetor de entrada do sistema é dado por u e o vetor de saída do sistema por y . Na representação de sistemas na forma de espaço de estados, a matriz \mathbf{D} representa a influência instantânea da entrada sobre a saída, ou seja, a que atua antes da variação dos estados. Na maior parte dos sistemas físicos, providos de inércia, esta influência é nula e, portanto, a matriz \mathbf{D} é nula.

3.4.1 Representação da Velocidade

A representação em espaço de estados para a velocidade o sistema apresenta duas variáveis de estado: velocidade w e torque τ .

Tendo em conta as equações 3.25 e 3.26, as matrizes do sistema em espaço de estados para a velocidade são:

$$\dot{\mathbf{x}} = \begin{bmatrix} \frac{dw(t)}{dt} \\ \frac{d\tau(t)}{dt} \end{bmatrix} \quad (3.29)$$

$$\mathbf{x} = \begin{bmatrix} w \\ \tau \end{bmatrix} \quad (3.30)$$

$$\mathbf{A}_v = \begin{bmatrix} -\frac{B_o}{J_o} & \frac{1}{J_o} \\ -\frac{K_m K_f}{L_a} & -\frac{R_a}{L_a} \end{bmatrix} \quad (3.31)$$

$$\mathbf{B}_v = \begin{bmatrix} 0 & -\frac{n}{J_o} \\ \frac{K_m}{L_a} & 0 \end{bmatrix} \quad (3.32)$$

$$\mathbf{C}_v = [1 \quad 0] \quad (3.33)$$

$$\mathbf{D}_v = [0] \quad (3.34)$$

$$(3.35)$$

$$\mathbf{u} = \begin{bmatrix} V \\ \tau_L \end{bmatrix}$$

$$y = w \cdot n \quad (3.36)$$

As matrizes \mathbf{A}_v , \mathbf{B}_v , \mathbf{C}_v e \mathbf{D}_v representam no seu conjunto o sistema a ser controlado também designado de sistema de posicionamento. O sinal que é aplicado à entrada do sistema é designado por sinal de entrada. Na representação anterior do sistema para espaço de estados, o sinal de controlo é dado pela tensão v . A carga a ser aplicada de forma externa ao sistema é dada por τ_L . Por fim, a velocidade w corresponde à variável de saída do sistema.

Para a carga τ_L nula, tem-se em conta a representação em espaço de estados anterior com a seguinte mudança na matriz B:

$$\mathbf{B}_{\tau_L=0} = \begin{bmatrix} 0 \\ \frac{K_m}{L_a} \end{bmatrix} \quad (3.37)$$

$$u = [v] \quad (3.38)$$

3.4.2 Representação da Posição

Na representação da posição do sistema contínuo na forma de espaço de estados é adicionada uma nova variável de estado, a posição θ . Considera-se a saída do sistema com a inclusão do sistema de engrenagens. Sendo assim, a saída do sistema é dada por $\theta \cdot n$.

$$\dot{\mathbf{x}} = \begin{bmatrix} \frac{d\theta(t)}{dt} \\ \frac{dw(t)}{dt} \\ \frac{d\tau(t)}{dt} \end{bmatrix} \quad (3.39)$$

$$\mathbf{A}_p = \begin{bmatrix} 0 & 1 & 0 \\ 0 & -\frac{B_o}{J_o} & \frac{1}{J_o} \\ 0 & -\frac{K_m K_f}{L} & -\frac{R_a}{L_a} \end{bmatrix} \quad (3.40)$$

$$\mathbf{B}_p = \begin{bmatrix} 0 & 0 \\ 0 & -\frac{n}{J_o} \\ \frac{K_m}{L_a} & 0 \end{bmatrix} \quad (3.41)$$

$$\mathbf{C}_p = [1 \quad 0 \quad 0] \quad (3.42)$$

$$\mathbf{D}_p = [0 \ 0] \quad (3.43)$$

De forma análoga à representação em espaço de estados da velocidade, considera-se que o motor não tem carga externa aplicada. Com isto, a matriz \mathbf{B} do sistema é a seguinte:

$$\mathbf{B}_p \tau_L=0 = \begin{bmatrix} 0 \\ 0 \\ \frac{K_m}{L_a} \end{bmatrix} \quad (3.44)$$

$$\mathbf{u} = [v] \quad (3.45)$$

3.5 Função de transferência do sistema no domínio discreto

A passagem do domínio contínuo para o domínio discreto é feita tendo em conta a presença de um dispositivo ZOH (*Zero-Order Hold*) e o intervalo de amostragem h .

A conversão do domínio contínuo para o domínio discreto é feita entre o plano s e o plano z . A função de transferência discreta na sua forma genérica [20] é dada pela seguinte expressão:

$$G(z) = \frac{b_1 z^{n-1} + b_2 z^{n-2} + \dots + b_n}{z^n + a_1 z^{n-1} + \dots + a_n} \quad (3.46)$$

A equação anterior refere-se a uma das formas de se representar uma função de transferência no domínio discreto. Partindo da equação 3.21, tem-se que a função de transferência para a velocidade em Laplace é equivalente a:

$$G_v(s) = \frac{ab}{(s+a)(s+b)} \quad (3.47)$$

A equação anterior é uma das equações que pode ser facilmente encontrada em tabelas de conversão entre o plano s e o plano z [21]. Recorrendo então a esta tabela, tem-se os seguintes coeficientes dependentes do intervalo de amostragem h :

$$a_1 = -(e^{-ah} + e^{-bh}) \quad (3.48)$$

$$a_2 = (e^{-(a+b)h}) \quad (3.49)$$

$$b_1 = \frac{b(1 - e^{-ah}) - a(1 - e^{-bh})}{b - a} \quad (3.50)$$

$$b_2 = \frac{a(1 - e^{-bh})e^{-ah} - b(1 - e^{-ah})e^{bh}}{b - a} \quad (3.51)$$

Os coeficientes apresentados anteriormente são aplicados na equação genérica 3.46, formando assim uma função de transferência discreta de segunda ordem.

Relativamente à função de transferência equivalente para a posição no domínio de Laplace, esta pode ser dada na forma:

$$G_p(s) = \frac{ab}{s(s+a)(s+b)} \quad (3.52)$$

A expressão apresentada anteriormente representa a função de transferência para a posição, que corresponde a um sistema de terceira ordem. Relativamente ao controlo da posição, pode-se considerar um sistema de segunda ordem, apresentando assim dois pólos. Se os pólos dominantes forem os pólos do integrador e da componente mecânica relativamente ao pólo da componente elétrica, pode-se desprezar este terceiro pólo. A razão pelo qual se despreza o terceiro pólo é pela pouca influência que este tem na resposta dinâmica do sistema. Isto só acontece se a localização do pólo não dominante for o mais à esquerda possível em relação aos pólos dominantes no lugar das raízes. Assim, a função de transferência da posição na forma simplificada é dada na forma:

$$G_p(s) = \frac{a}{s(s+a)} \quad (3.53)$$

O modelo discreto da posição na forma de função de transferência é obtido usando os seguintes coeficientes, que dependem do intervalo de amostragem h [21]:

$$a_1 = -(1 + e^{-ah}) \quad (3.54)$$

$$a_2 = e^{-ah} \quad (3.55)$$

$$b_1 = \frac{1}{a}(ah - 1 + e^{-ah}) \quad (3.56)$$

$$b_2 = \frac{1}{a}(1 - e^{-ah} + ahe^{-ah}) \quad (3.57)$$

3.6 Representação em espaço de estados no domínio discreto

Neste tópico determina-se o equivalente discreto de um sistema contínuo em espaço de estados. Considerando um circuito de retenção (sistema de ordem zero - ZOH), tem-se a seguinte representação na forma de espaço de estados discreta [22],

$$\mathbf{x}(kh + h) = \mathbf{G}\mathbf{x}(kh) + \mathbf{H}u(kh) \quad (3.58)$$

$$y(kh) = \mathbf{C}\mathbf{x}(kh) + \mathbf{D}u(kh) \quad (3.59)$$

Sendo que a matriz \mathbf{D} é nula, tem-se

$$y(kh) = \mathbf{C}\mathbf{x}(kh) \quad (3.60)$$

A equação 3.58 representa a equação de estado de um sistema discreto amostrado com um período de amostragem h e a equação 3.59 representa a equação de saída de um sistema discreto com um período de amostragem h . Das expressões 3.58 e 3.59 define-se as matrizes \mathbf{G} e \mathbf{H} da seguinte forma [22]:

$$\mathbf{G} = \Phi(h) = e^{A h} \quad (3.61)$$

$$\mathbf{H} = \Gamma(h) = \int_0^h e^{A q} \mathbf{B} dq \quad (3.62)$$

Simplificando a equação 3.62,

$$\Gamma(h) = \left(\int_0^h e^{A q} dq \right) \mathbf{B} = (e^{A h} - \mathbf{I}) \mathbf{A}^{-1} \mathbf{B} \quad (3.63)$$

3.6.1 Solução da Equação de Estado no domínio discreto

A solução da equação de estados permite calcular o valor da saída y para os vários instantes de tempo. Partindo da equação 3.58 e considerando o período de amostragem $h = 1$, obtém-se a seguinte equação para o instante $k + 1$:

$$x(k + 1) = \mathbf{G}x(k) + \mathbf{H}u(k) \quad (3.64)$$

Para o instante $k + 2$, tem-se a seguinte equação:

$$x(k + 2) = \mathbf{G}x(k + 1) + \mathbf{H}u(k + 1) \quad (3.65)$$

Substituindo a equação 3.64 na equação 3.65,

$$x(k + 2) = \mathbf{G}^2 x(k) + \mathbf{G} \mathbf{H} u(k) + \mathbf{H} u(k + 1) \quad (3.66)$$

Pelas equações 3.64, 3.65 e 3.66 é possível obter a solução genérica da equação de estado no domínio discreto.

$$x(k) = \mathbf{G}^k x(0) + \sum_{i=0}^{k-1} \mathbf{C} \mathbf{G}^{k-i-1} \mathbf{H} u(i) \quad (3.67)$$

em que $x(0)$ representa o estado inicial. Por fim, a solução genérica da equação de saída no domínio discreto é dada por:

$$y(k) = \mathbf{C} \mathbf{G}^k x(0) + \sum_{i=0}^{k-1} \mathbf{C} \mathbf{G}^{k-i-1} \mathbf{H} u(i) \quad (3.68)$$

3.7 Seleção do intervalo de amostragem h

De acordo com o tópico anterior, onde se apresenta a metodologia utilizada para calcular as matrizes do sistema na forma discreta, é necessário escolher o intervalo de amostragem h .

Existem várias formas de escolher o período de amostragem. Uma delas é pelo teorema de Nyquist-Shannon. No domínio do processamento digital de sinal, este teorema aplica-se quando se pretende amostrar um sinal analógico. O processo de amostragem consiste na transformação de um sinal analógico num sinal digital.

Segundo o Teorema de Nyquist-Shannon, para que não haja perda de informação no sinal amostrado é necessário que a frequência de amostragem utilizada seja pelo menos duas vezes superior ou igual à frequência máxima do sinal original.

Para a discretização de sistemas contínuos, é utilizado um outro critério de seleção do intervalo de amostragem h . Segundo Åström [20], o período de amostragem deve estar num valor entre quatro a dez amostras do tempo de subida T_r do sistema, isto é,

$$\frac{T_r}{10} \leq h \leq \frac{T_r}{4} \quad (3.69)$$

Capítulo 4 Simulações do Sistema

No presente capítulo, é efetuada a simulação do sistema de posicionamento modelado no capítulo 3 com a implementação do controlador de posição. De forma paralela à simulação do sistema são efetuadas as respetivas análises ao comportamento do sistema de controlo de posição.

Sendo o objetivo o controlo do sistema de posicionamento utilizando um modelo de espaço de estados, o ponto de partida na simulação é o sistema em malha aberta, seguindo-se o sistema com realimentação de estados sem ação integral. Por fim, é implementado o sistema com ação integral e são efetuados testes com a introdução de ruído nas malhas de realimentação e com a variação de carga. Todo o processo de modelação, simulação e análise do sistema de posicionamento, é efetuado numa abordagem contínua e combinada. É utilizada a ferramenta de simulação MATLAB/Simulink da empresa Mathworks [23].

4.1 Descrição do método combinado

Como já referido anteriormente, neste capítulo simula-se o sistema em modo contínuo e em modo combinado. É utilizado o método combinado para simular e analisar o sistema de controlo de posição de forma a que se faça um controlo digital do mesmo. Na utilização deste método, o sistema a controlar é interligado ao controlador. A figura seguinte mostra a interligação entre o sistema a controlar e o controlador.

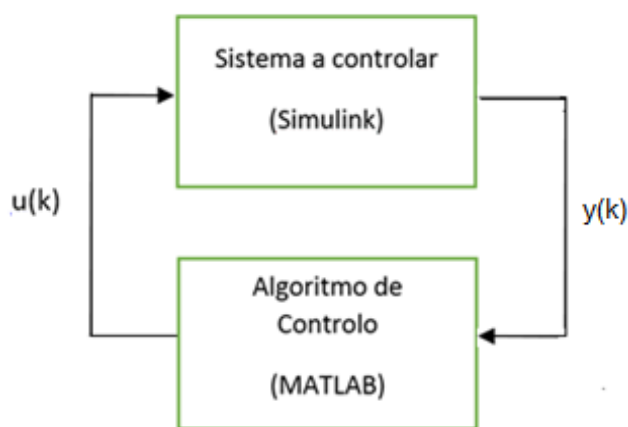


Figura 19 - Interligação entre o sistema a controlar e o controlador

Na simulação combinada, o sistema a controlar refere-se ao sistema de posicionamento, que é implementado em Simulink na forma de espaço de estados no domínio contínuo. O algoritmo de controlo é implementado em MATLAB.

O sistema contínuo é simulado da seguinte forma: por cada instante de amostragem, o algoritmo de controlo implementado no MATLAB envia para o Simulink um sinal de entrada $u(k)$ de modo a que o sistema seja simulado durante um período de amostragem com base nos dados passado e no sinal novo de entrada. O resultado da simulação do sistema contínuo é um sinal cuja amostra atual corresponde a $y(k)$ [24].

Uma das vantagens de utilização desta abordagem é o fato da simulação do algoritmo de controlo ser independente da simulação do sistema a controlar. Com esta abordagem é

possível verificar o efeito da escolha do período de amostragem. É possível alterar o algoritmo de controlo de uma forma simples sem que isso dependa do sistema a controlar [24]. A grande desvantagem da utilização deste método é o fato de a simulação ser mais lenta.

4.2 Parâmetros do Sistema

Para efeitos de simulação do sistema representado em espaço de estados, consideram-se os parâmetros do conjunto motor DC, sistema de engrenagens e carga. Na tabela seguinte apresentam-se os parâmetros do sistema.

Resistência da armadura do motor	$R_a = 2 \Omega$
Indutância da armadura do motor	$L_a = 0.5 \text{ H}$
Momento de inércia do conjunto motor, engrenagens e carga	$J_o = 0.045 \text{ kg.m}^2$
Coeficiente de atrito do conjunto motor, engrenagens e carga	$B_o = 0.11 \text{ N.m.s}$
Constante de torque do motor	$K_m = 0.3 \text{ N.m/A}$
Constante de força contra-eletromotriz	$K_f = 0.3 \text{ N.m/A}$
Relação das engrenagens	$n = 0.5$

Tabela 2 - Parâmetros do Sistema

4.3 Função de Transferência do Sistema

Neste tópico apresentam-se as funções de transferência que representam a velocidade e a posição do sistema referenciado no capítulo anterior na Figura 18. Numa fase inicial apresenta-se o modelo do sistema por função de transferência, sendo posteriormente convertido para um modelo de espaço de estados.

4.3.1 Função de Transferência da velocidade no domínio contínuo

Utilizando os parâmetros considerados no tópico 4.2, apresenta-se a seguinte função de transferência, que representa a velocidade do sistema, de acordo com a equação 3.21:

$$G_{11}(s) = \frac{0.15}{0.0225s^2 + 0.1462s + 0.315} \quad (4.1)$$

A função de transferência representada na expressão 4.1 está na forma da função de transferência da velocidade apresentada em 3.21, sendo esta de segunda ordem. Os pólos do sistema em malha aberta estão localizados em $-3.04 \pm 2.1j$. A discretização da função de transferência da velocidade do sistema é apresentada no tópico seguinte.

4.3.2 Função de Transferência da velocidade no domínio discreto

A obtenção da função de transferência discretizada da velocidade é dependente da seleção do intervalo de amostragem h . Para a seleção do intervalo de amostragem, utiliza-se o critério apresentado no tópico 3.7, referente ao tempo de subida na resposta do sistema no

domínio contínuo. Aplicando um degrau unitário ao sistema representado na equação 4.1, obtém-se o seguinte resultado.

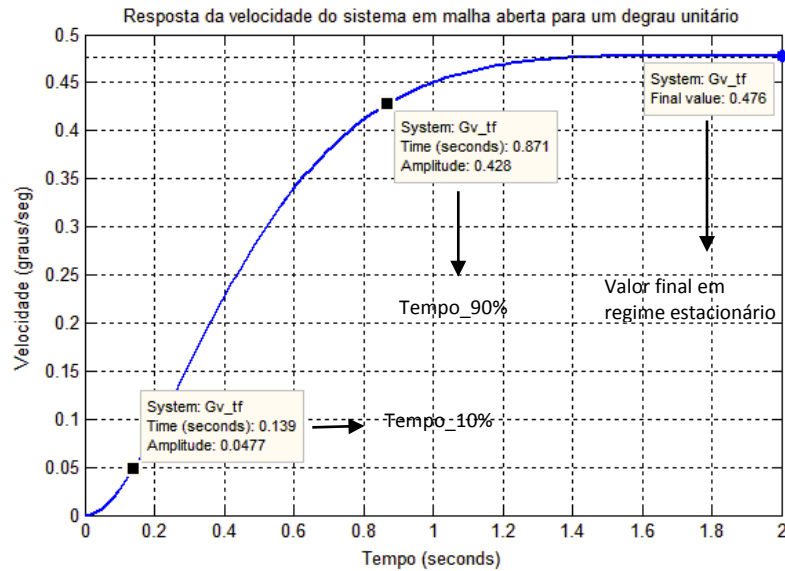


Figura 20 - Resposta do Sistema em malha aberta para um degrau unitário

O tempo de subida (*rise time*) corresponde à diferença entre o tempo em que é atingido 90% do valor em regime estacionário e o tempo de 10% do valor final. Sendo assim, o tempo de subida obtido na resposta da velocidade do sistema da Figura 20 é aproximadamente 730 ms. Utilizando o critério referenciado na equação 3.69, obtém-se um intervalo de amostragem de 183ms para 4 amostras.

Considerando o intervalo de amostragem h de valor 180ms, a função de transferência para a velocidade em função da inversa de z é:

$$G_{11}(z^{-1}) = \frac{0.07321 z^{-1} + 0.04947 z^{-2}}{1 - 1.053 z^{-1} + 0.3104} \quad (4.2)$$

Escrevendo a equação anterior em função de z , tem-se:

$$G_{11}(z) = \frac{0.07321 z + 0.04947}{z^2 - 1.053 z + 0.3104} \quad (4.3)$$

4.3.3 Função de Transferência da posição no domínio contínuo

Como a posição é o integral da velocidade e considerando a equação 4.1 referente à velocidade, tem-se a seguinte expressão:

$$G_{21}(s) = \frac{0.15}{0.0225s^2 + 0.1462s + 0.315} \cdot \frac{1}{s} \quad (4.4)$$

Simplificando a equação 4.4, obtém-se a seguinte função de transferência para a posição:

$$G_{21}(s) = \frac{0.15}{0.0225s^3 + 0.1462s^2 + 0.315s} \quad (4.5)$$

4.3.4 Função de Transferência da posição no domínio discreto

Considerando o mesmo intervalo de amostragem (180ms), obteve-se a seguinte função de transferência discreta:

$$G_{21}(z^{-1}) = \frac{0.004854 z^{-1} + 0.01453 z^{-2} + 0.0027z^{-3}}{1 - 2.053 z^{-1} + 1.363 z^{-2} - 0.3104z^{-3}} \quad (4.6)$$

Colocando a equação anterior em função de z, tem-se:

$$G_{21}(z) = \frac{0.004854 z^2 + 0.01453 z + 0.0027}{z^3 - 2.053 z^2 + 1.363 - 0.3104} \quad (4.7)$$

Nesta seção representou-se o sistema de posicionamento na forma de função de transferência no domínio contínuo e discreto.

De seguida apresenta-se o sistema na forma de espaço de estados com os valores dos parâmetros considerados na seção 4.2.

4.4 Sistema com malha aberta em Espaço de Estados

4.4.1 Domínio Contínuo

Para efeitos de simulação, considera-se apenas o modelo do sistema para a posição para posterior implementação de um controlador por realimentação de estados.

Nesta seção é apresentado o modelo do sistema a controlar na sua forma mais básica. A figura seguinte mostra a representação do sistema em malha aberta na forma de espaço de estados no domínio contínuo.

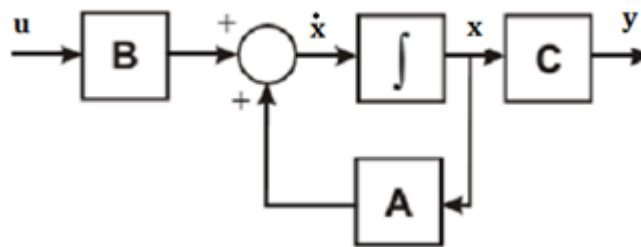


Figura 21 - Representação do sistema em Malha Aberta no domínio contínuo [25]

A Figura 21 representa o sistema a controlar em malha aberta, de acordo com o modelo de posição obtido na seção 3.4.2. De referir que o sistema a controlar é representado pelo modelo na forma de espaço de estados. Tendo em conta a função de transferência que representa a posição do sistema, a conversão para o modelo em espaço de estados da Figura

21 é feita utilizando as funcionalidades do MATLAB. Sendo assim, obtém-se a seguinte representação da posição do sistema em espaço de estados:

$$A_p = \begin{bmatrix} 0 & 1 & 0 \\ 0 & -2.5 & 22.2 \\ 0 & -0.18 & -4 \end{bmatrix} \quad (4.8)$$

$$B_p_{\tau_L=0} = \begin{bmatrix} 0 \\ 0 \\ 0.6 \end{bmatrix} \quad (4.9)$$

$$C_p = [0.5 \quad 0 \quad 0] \quad (4.10)$$

4.4.1.1 Construção do Diagrama de Blocos em Simulink

Para a simulação e análise do comportamento dinâmico do sistema de posicionamento em malha aberta, procedeu-se à sua modelação em Simulink, utilizando como sinal de entrada um sinal em forma de degrau.

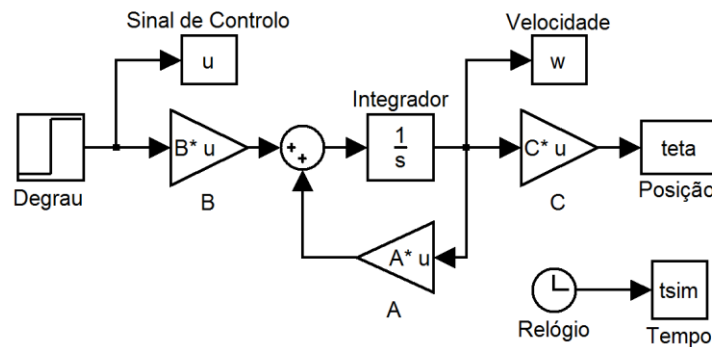


Figura 22 - Diagrama de blocos do sistema em malha aberta com entrada em degrau

O diagrama de blocos da Figura 22 é constituído pelo sistema a controlar referenciado na Figura 21 e pelos blocos que representam o sinal de controlo, velocidade, posição e tempo. Estes blocos são designados por blocos *workspace* [26] e permitem enviar os dados obtidos da simulação diretamente para o *workspace* do MATLAB e, portanto, construir os gráficos com a resposta da dinâmica do sistema de posicionamento. Para o modelo em malha aberta, a entrada do sistema corresponde a um valor de tensão, em volts e a saída do mesmo corresponde a uma posição, expressa em graus.

4.4.1.2 Resposta do sistema a um degrau de referência

Definiu-se um tempo de simulação de 40 segundos e para o sinal de entrada um sinal em forma de degrau com um nível de excitação, expresso em *volts*.

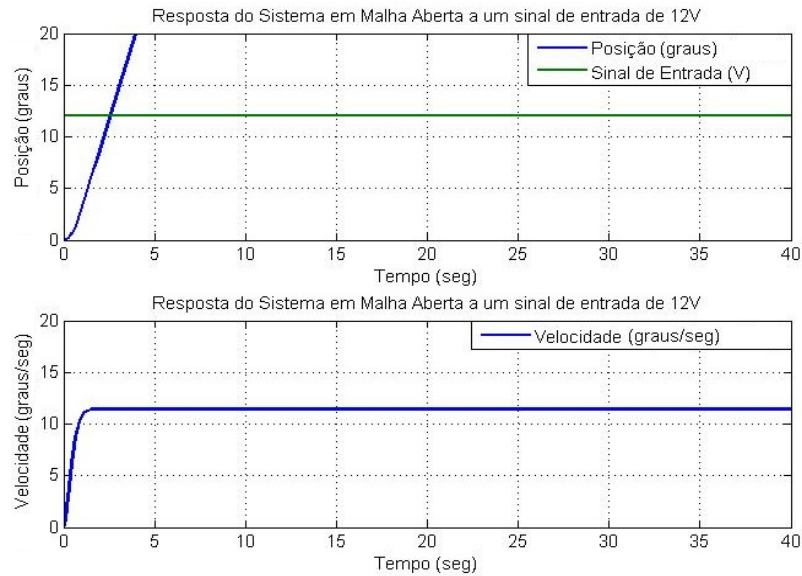


Figura 23 - Resposta do Sistema contínuo em malha aberta a um degrau de referência de 12V

Por observação da figura anterior, verifica-se que não há qualquer tipo de controlo sob a posição do sistema, isto é, ao longo do tempo esta cresce sem seguir qualquer tipo de referência. Relativamente à velocidade, verifica-se que, com o decorrer do tempo, cresce atingindo um valor constante diferente de zero.

4.4.2 Domínio Discreto

No tópico anterior, definiu-se o modelo contínuo do sistema e analisou-se a resposta do mesmo em malha aberta. Neste tópico, é apresentado o modelo do sistema de posicionamento na sua forma discreta e de seguida analisa-se a resposta do sistema na forma discreta.

O modelo discreto do sistema descreve a relação entre a entrada e a saída do sistema em instantes de tempo discretos, cujo intervalo é definido pelo período de amostragem. Para obtenção das matrizes do sistema na forma discreta, utiliza-se a metodologia apresentada no tópico 3.6.

4.4.2.1 Modelo Discreto do Sistema

O sistema a controlar na sua forma discreta é dado pela figura seguinte.

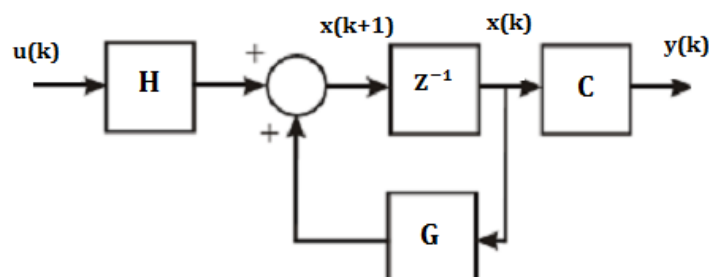


Figura 24 - Representação do sistema em malha aberta no domínio discreto [22]

As matrizes discretas do sistema representado na Figura 24 podem ser obtidas considerando as equações referenciadas no tópico 3.6 e o intervalo de amostragem selecionado utilizando o critério apresentado no tópico 3.7. O modelo discreto da posição do sistema representado em espaço de estados é composto pelas matrizes \mathbf{G} , \mathbf{H} e \mathbf{C} , conforme se verifica na figura anterior. Sendo assim, as matrizes \mathbf{G} e \mathbf{H} são dadas pelas seguintes expressões.

$$\mathbf{G} = \mathbf{G}_p = \begin{bmatrix} 1 & 0.142 & 0.244 \\ 0 & 0.600 & 2.187 \\ 0 & -0.017 & 0.453 \end{bmatrix} \quad (4.11)$$

$$\mathbf{H} = \mathbf{H}_p = \begin{bmatrix} 0.009 \\ 0.146 \\ 0.075 \end{bmatrix} \quad (4.12)$$

4.4.2.2 Diagrama de Blocos em Simulink

Neste tópico apresenta-se o diagrama de blocos referente ao modelo discreto da posição na forma de espaço de estados, implementado em Simulink.

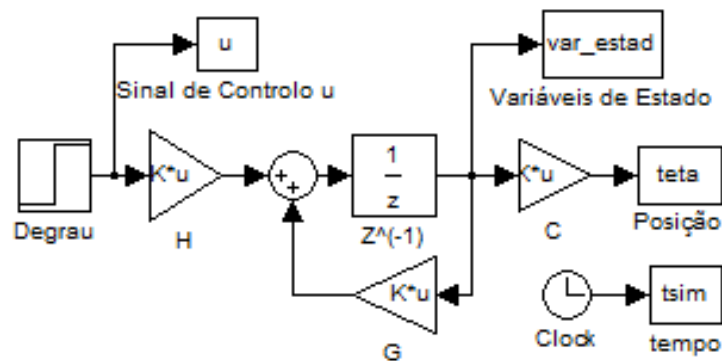


Figura 25 - Diagrama de blocos do sistema discreto em malha aberta com entrada em degrau

Comparativamente ao diagrama de blocos do modelo contínuo apresentado na Figura 22, o diagrama da figura anterior apresenta diferenças nos blocos das matrizes \mathbf{G} , \mathbf{H} e \mathbf{C} do sistema e também no bloco integrador, sendo representado por um bloco *unit delay* [26].

4.4.2.3 Resposta do sistema discreto a um degrau de referência

Considerando as mesmas condições de simulação do tópico 4.4.1.2, onde é efetuada a simulação do mesmo sistema em modo contínuo, foram efetuadas as mesmas simulações para o modelo discreto. O resultado da resposta do sistema representado pelo modelo discreto para o sinal de entrada referido é dado pela figura seguinte.

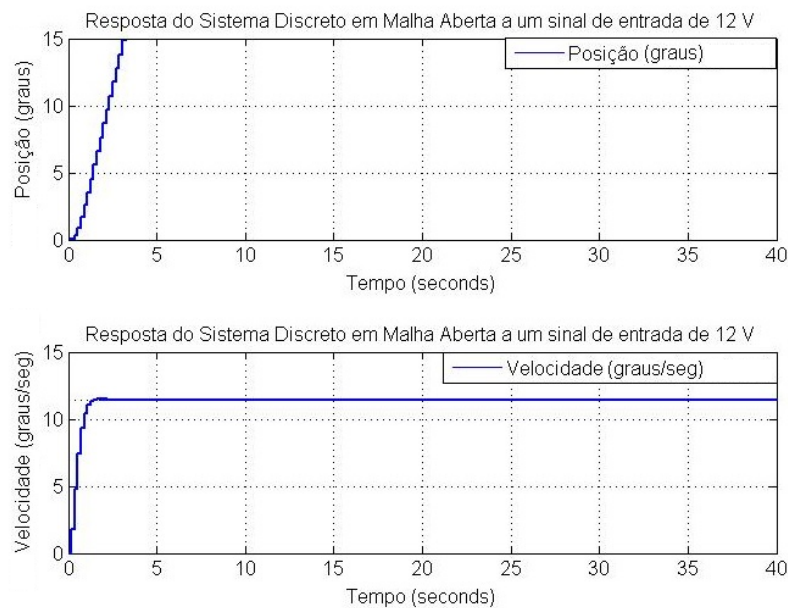


Figura 26 - Resposta do sistema discreto em malha aberta a um sinal de referência de 12V

Utilizando o modelo discreto da posição em espaço de estados e simulando o sistema em malha aberta, obteve-se o resultado da figura anterior. Como era de esperar, não existe qualquer tipo de controlo sob o sistema, visto que o sistema se encontra em malha aberta.

4.4.3 Simulação Combinada

4.4.3.1 Diagrama de Blocos do Sistema a controlar

Para efetuar a simulação em modo combinado, implementou-se em Simulink o modelo do sistema a controlar que representa o sistema a controlar no modo contínuo. O modelo do sistema a controlar serve de base a todas as simulações em modo combinado que se efetuam nos tópicos seguintes.

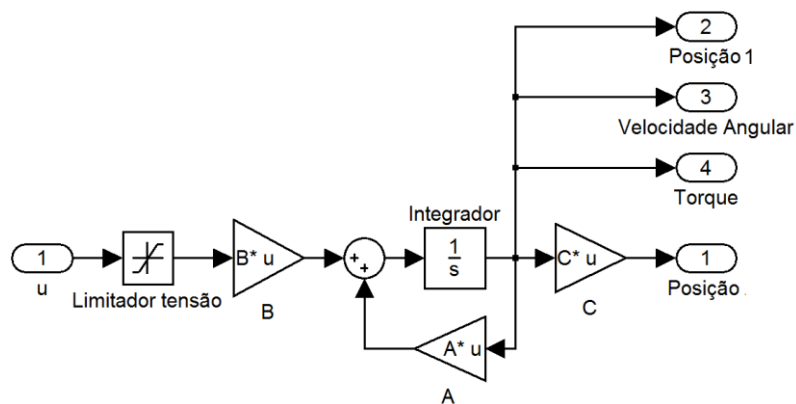


Figura 27 - Diagrama de blocos do sistema a controlar

O diagrama de blocos da Figura 27 é constituído pelo sistema a controlar na forma de espaço de estados representado através das matrizes A , B e C em modo contínuo, referenciadas no tópico 4.4.1. Utilizaram-se blocos de entrada e de saída de forma a que o sistema contínuo seja simulado com interligação ao algoritmo de controlo, desenvolvido em MATLAB. O bloco de entrada corresponde ao bloco u , que representa o sinal de entrada do sistema. Os blocos que representam as variáveis de estado correspondem aos blocos 2, 3 e 4. A saída do sistema corresponde à posição, representada pelo bloco 1. É assumido que neste sistema as variáveis de estado da velocidade e do torque estão disponíveis. Por fim, foi utilizado um bloco que limita a tensão à entrada do sistema. Foi definido um limite de tensão de $\pm 60V$.

4.5 Sistema com Realimentação em Espaço de Estados

No tópico 4.4 foi modelado e simulado o sistema de posicionamento em espaço de estados na sua forma mais básica, isto é, em malha aberta. O objetivo principal é modelar o sistema de modo a que este se comporte como um seguidor de referência. Para isso é necessário a apresentação de um sistema que tenha uma malha de realimentação de estados.

4.5.1 Classificação de Sistemas com Realimentação de Estados

Entre as várias classificações que se podem fazer dos sistemas de controlo, salienta-se a que divide os sistemas em dois grupos: reguladores de estado e sistemas servo.

Um regulador de estados é usado para manter a sua saída numa condição fixa de operação. No domínio de espaço de estados, a saída do sistema não acompanha a entrada do mesmo, ou seja, o sistema não apresenta qualquer entrada de referência. Segundo [19], o regulador de estados na forma de espaço de estados é dado pela figura seguinte.

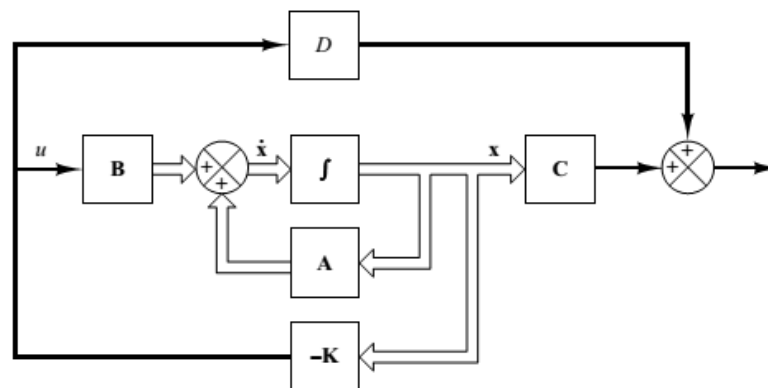


Figura 28 - Regulador de estados na forma de espaço de estados [19]

Um sistema servo é idêntico a um regulador de estados com a diferença que a saída do sistema acompanha a entrada. A figura seguinte é representativa de um sistema servo na forma de espaço de estados [19].

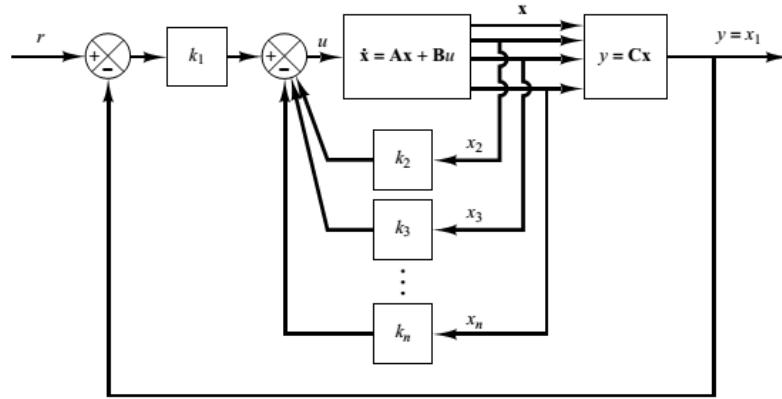


Figura 29 - Sistema servo na forma de espaço de estados [19]

A Figura 29 mostra a configuração de um sistema servo quando o sistema a controlar possui um integrador. Um sistema de controle de posição apresenta a mesma configuração da figura anterior. Por observação da figura anterior, a saída do sistema que corresponde à posição acompanha o sinal de referência r . A metodologia utilizada para calcular as matrizes de realimentação de estados é apresentada de seguida. O sinal de controlo u é dado pela seguinte equação:

$$u = -[0 \quad k_2 \quad k_3 \quad \dots \quad k_n] \begin{bmatrix} x_1 \\ x_2 \\ \cdot \\ \cdot \\ x_n \end{bmatrix} + k_1(r - x_1) \quad (4.13)$$

$$u = -\mathbf{K}\mathbf{x} + k_1 r \quad (4.14)$$

A equação da dinâmica do sistema (equação de estado) é dada por:

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}u = (\mathbf{A} - \mathbf{B}\mathbf{k})\mathbf{x} + \mathbf{B}k_1 r \quad (4.15)$$

4.5.2 Domínio Contínuo

Tendo sido apresentada anteriormente a configuração em espaço de estados do sistema servo, nesta seção é descrita toda a metodologia de cálculo das matrizes de realimentação de estados e, por fim, efetua-se a simulação do sistema em modo contínuo.

4.5.2.1 Controlabilidade

A controlabilidade é uma propriedade de enorme relevância no projeto de sistemas de controlo. Um sistema é controlável se for possível, por controlo da entrada, conduzir os estados de um valor inicial a qualquer valor final, num intervalo de tempo finito [19]. Matematicamente, um sistema possui estados completamente controláveis se e só se a característica da matriz de controlabilidade for igual ao número de estados n do sistema, de acordo com a seguinte equação:

$$caraterística_matricial([B \quad AB \quad A^2B \quad \dots \quad A^{n-1}B]) = n \quad (4.16)$$

Para analisar a controlabilidade do sistema, utilizaram-se funções pertencentes à *Control System Toolbox* do MATLAB. Sendo assim, o valor do posto matricial obtido foi de 3, que corresponde ao número de variáveis de estado do sistema, o que permite concluir que o sistema é totalmente controlável.

4.5.2.2 Determinação dos pólos da malha fechada

O processo de determinação dos ganhos de realimentação de estados pressupõe que se definam os pólos desejados para a malha fechada, sendo posteriormente utilizados no cálculo da matriz de realimentação de estados pelo método de *Ackermann* [19]. A escolha dos pólos desejados para a malha fechada baseia-se nos requisitos da resposta transitória e/ou resposta em frequência. Na presente dissertação, o sistema foi modelado em espaço de estados e a partir dele foram especificados os pólos recorrendo aos requisitos da resposta transitória. Em muitos casos práticos, as características do desempenho desejadas para os sistemas de controlo são especificadas em termos de parâmetros no domínio do tempo. De uma forma geral, as características de desempenho de um sistema são especificadas em termos de resposta transitória para uma entrada de degrau unitário.

As características que contribuem para a especificação da resposta transitória de um sistema a uma entrada forma de degrau são [25]: tempo de subida (*rise time*), tempo de atraso (*delay time*), tempo de estabelecimento e sobrelevação.

Relativamente ao modelo em espaço de estados para a posição do sistema referenciado no tópico 4.4.1, este corresponde a um modelo do sistema de terceira ordem com as raízes localizadas no seguinte diagrama.

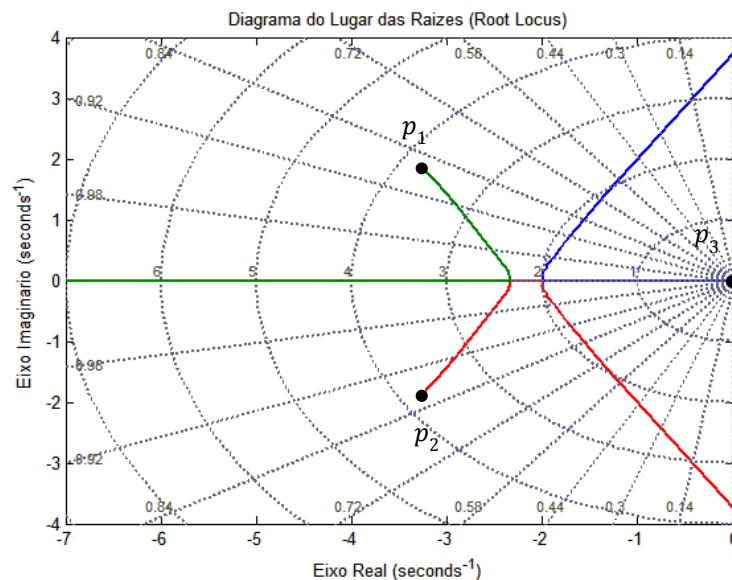


Figura 30 - Diagrama do Lugar das raízes do modelo para a posição do sistema contínuo

Por observação da figura anterior, o modelo do sistema de posicionamento que é de terceira ordem apresenta dois pólos complexos e um pólo real. Os dois pólos complexos, correspondentes à componente elétrica e mecânica encontram-se localizados no plano s em

$p_1 = -3.04 + 2.1j$ e $p_2 = -3.04 - 2.1j$, respectivamente. O terceiro pólo, correspondente ao pólo integrador é um pólo real com a localização no plano s em $p_3 = 0$.

A determinação dos pólos desejados para a malha fechada é feita considerando o modelo do sistema de segunda ordem e dois parâmetros denominados de frequência natural e fator de amortecimento. Segundo [25], os dois parâmetros referenciados anteriormente são utilizados para descrever as características da resposta transiente do sistema. A frequência natural ω_n do sistema de segunda ordem corresponde à frequência de oscilação do sistema sem amortecimento. O fator de amortecimento, dado pelo símbolo ξ , é um parâmetro que define o tipo de amortecimento do sistema na sua fase transitória. A figura seguinte mostra a resposta transitória de um sistema de segunda ordem com vários fatores de amortecimento.

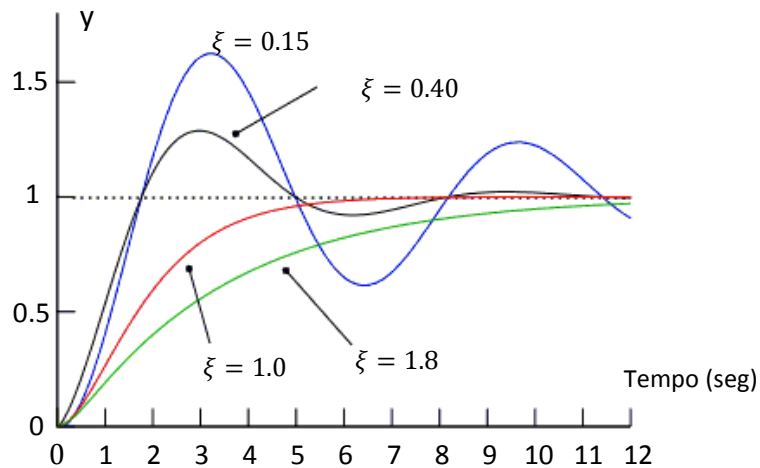


Figura 31 - Resposta transitória de um sistema de segunda ordem

Um sistema pode ser classificado de sobre-amortecido para $\xi > 1$, criticamente amortecido para $\xi = 1$ e sub-amortecido para $0 < \xi < 1$ [25]. Para um sistema sub-amortecido, o fator de amortecimento e a frequência natural são dados pelas equações 4.17 e 4.18, respectivamente:

$$\xi = \frac{-\ln(\%OS / 100)}{\sqrt{\pi^2 + \ln^2(\%OS / 100)}} \quad (4.17)$$

$$\omega_n = \frac{4}{\xi T_s} \quad (4.18)$$

Nas expressões apresentadas, o parâmetro OS refere-se à percentagem de sobrelevação (*overshoot*) e o parâmetro T_s representa o tempo de estabelecimento (*settling time*). O tempo de estabelecimento é o tempo necessário para que a resposta do sistema alcance e permaneça dentro de uma gama definida em termos do valor final no regime estacionário (normalmente 2% ou 5%). A expressão 4.18 refere-se ao valor do tempo de estabelecimento, dado por 2% do valor final em regime estacionário.

Definiu-se um valor de sobrelevação de 10% e um tempo de estabelecimento de 3 segundos para determinação dos pólos desejados para a malha fechada. Sendo assim, obteve-se um fator de amortecimento de 0.59 e uma frequência natural de 2.25 rad/seg. Os pólos

desejados para a malha fechada são dois pólos complexos de valor $-1.33 \pm 1.81j$. O pólo não dominante, que corresponde ao pólo elétrico é igual a cinco vezes a componente real dos pólos dominantes [16], sendo neste caso posicionado em -6.6. Sendo assim, o pólo elétrico tem pouca influência na resposta da dinâmica do sistema.

4.5.2.3 Determinação da Matriz de Realimentação de Estados

Uma vez calculados os pólos desejados para a malha fechada que constitui o sistema de controlo de posição, segue-se a etapa da determinação da matriz de realimentação de estados. Um dos métodos bastante utilizado para este fim é o método de *Ackermann*. De acordo com [19], a matriz de realimentação de estados é dada por:

$$\mathbf{K} = [k_1 \ k_2 \ \dots \ k_n] \quad (4.19)$$

A equação 4.19 apresenta o resultado final dos vários ganhos de realimentação, que formam a matriz de realimentação. No entanto, a seguinte equação mostra de forma detalhada a fórmula de *Ackermann*.

$$\mathbf{K} = [0 \ 0 \ \dots \ 1] \mathbf{P}_c^{-1} \mathbf{q}(\mathbf{A}) \quad (4.20)$$

A matriz \mathbf{P}_c representa a matriz de controlabilidade, já referenciada anteriormente no tópico 4.5.2.1. A equação característica do sistema com os pólos desejados para a malha fechada é:

$$q(s) = s^n + \alpha_1 s^{n-1} + \dots + \alpha_n \quad (4.21)$$

De acordo com a equação 4.21,

$$q(\mathbf{A}) = \mathbf{A}^n + \alpha_1 \mathbf{A}^{n-1} + \dots + \alpha_n \quad (4.22)$$

Utilizando esta metodologia para calcular a matriz de realimentação do sistema de posicionamento e considerando os pólos desejados para a malha fechada determinados no tópico anterior, tem-se que:

$$q(s) = (s + 1.33 - 1.81j)(s + 1.33 + 1.33j)(s + 6.6) \quad (4.23)$$

Simplificando a equação 4.23,

$$q(s) = s^3 + 9.33s^2 + 22.86s + 33.9 \quad (4.24)$$

Colocando a equação 4.24 na forma representada pela equação 4.22,

$$\mathbf{q}(\mathbf{A}) = \mathbf{A}^3 + 9.33\mathbf{A}^2 + 22.86\mathbf{A} + 33.9 \quad (4.25)$$

Por fim, obteve-se a matriz de realimentação de estados. Dado que a ordem do sistema é $n=3$, a matriz de realimentação é constituída pelos elementos de ganho k_1 , k_2 e k_3 . Obteve-se a seguinte matriz de realimentação:

$$K = [2.54 \quad 0.13 \quad 4.72] \quad (4.26)$$

4.5.2.4 Construção do Sistema Servo em Simulink

A figura seguinte mostra o diagrama de blocos implementado em Simulink do sistema servo, já apresentado anteriormente no tópico 4.5.1.

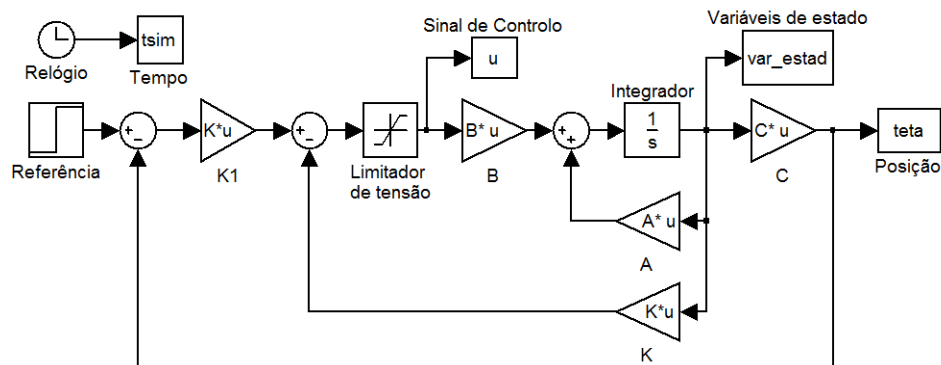


Figura 32 - Diagrama de blocos em Simulink do sistema servo sem ação integral

Considerando os ganhos de realimentação de estados obtidos na equação 4.26 e aplicando estes às matrizes de ganho K e K_1 da Figura 32 tem-se:

$$K = [0 \quad 0.13 \quad 4.72] \quad (4.27)$$

$$K_1 = [2.54] \quad (4.28)$$

O diagrama de blocos apresentado anteriormente apresenta os mesmos parâmetros utilizados nas simulações anteriores com a diferença que neste caso utiliza um bloco designado de degrau para a posição de referência e os blocos para as matrizes de realimentação.

O tópico seguinte refere-se aos resultados obtidos da simulação do sistema servo para uma posição de referência, apresentada em graus e posterior análise das variáveis de estado e do sinal de controlo.

4.5.2.5 Resposta do sistema a uma posição de referência

Relativamente às condições de simulação do sistema servo, considerou-se uma posição de referência de 40 graus. Em seguida apresenta-se o resultado do comportamento da posição do sistema, bem como as variáveis de estado e o sinal de controlo para um tempo de simulação de 60 segundos.

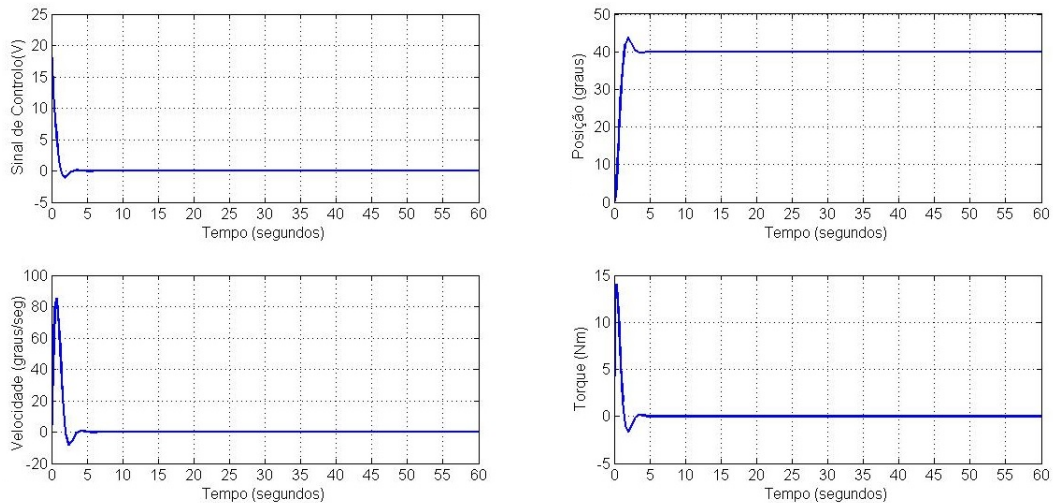


Figura 33 - Resposta do sistema servo em modo contínuo para uma referência de 40 graus

Por observação do resultado da Figura 33, o sistema servo apresenta um comportamento típico de um seguidor de referência, quando lhe é aplicada na entrada uma posição de referência.

Na simulação efetuada, o sistema servo atinge o estado estacionário em cerca de 6 segundos. A determinação dos pólos do sistema para a malha fechada é determinante neste resultado. Um dos parâmetros considerados para a determinação dos pólos da malha fechada foi o tempo de estabelecimento de 3 segundos. Com isto, estes resultados são válidos uma vez que o sistema atinge 98% do seu valor final em menos de 3 segundos, que corresponde ao tempo de estabelecimento.

No instante inicial da resposta, a velocidade do sistema dada em graus/segundo atinge um valor de pico e à medida que o sistema servo se aproxima da posição de referência, a velocidade tende para o seu valor nulo. O mesmo acontece com a variável de estado torque (N.m). Relativamente ao sinal de controlo, este tende para 0V à medida que o sistema se aproxima do estado estacionário.

4.5.3 Simulação Combinada

Como referido no tópico 4.1, a simulação combinada baseia-se na interligação de um sistema a controlar em modo contínuo (implementado em Simulink) com um controlador discreto (em MATLAB). Uma vez que se trata de um sistema com realimentação de estados, o primeiro passo é proceder à discretização da matriz de realimentação. O processo de discretização da matriz de realimentação de estados é apresentado em seguida e, posteriormente são apresentados os resultados da simulação em modo combinado.

4.5.3.1 Discretização da Matriz de Realimentação de Estados

A obtenção da matriz de realimentação de estados em modo combinado é efetuada de acordo com o método utilizado para o modo contínuo e apresentado anteriormente no tópico 4.5.2.2. Sendo o objetivo a discretização da matriz de realimentação, obtém-se em primeiro lugar os pólos do sistema no plano Z.

A discretização dos pólos utilizando um dispositivo ZOH é feita através de uma transformação que envolve uma exponencial. A relação entre os pólos da transformada de Laplace obtidos no tópico 4.5.2.2 e os pólos do sistema no plano Z é dada pela seguinte expressão [27].

$$p_z = e^{p_s T} \quad (4.29)$$

Na equação anterior, os pólos no plano Z são dados pela variável p_z enquanto que os pólos no domínio de Laplace são representados pela variável p_s . A terceira variável refere-se ao intervalo de amostragem T . Considerando os pólos da Transformada de Laplace obtidos no tópico 4.5.2.2 e o intervalo de amostragem de 180ms, os pólos pertencentes ao plano Z são:

$$\mathbf{p}_z = [p_{z1} \ p_{z2} \ p_{z3}] \quad (4.30)$$

$$\mathbf{p}_z = [0.74 + 0.25i \ 0.74 - 0.25i \ 0.3] \quad (4.31)$$

Uma vez que todos os pólos no plano Z estão localizados no interior do círculo unitário, a condição de estabilidade é cumprida. O cálculo da matriz de realimentação de estados baseia-se no método de *Ackermann*, já utilizado anteriormente na seção 4.5.2.3. De salientar que são utilizadas as matrizes \mathbf{G} e \mathbf{H} em lugar das matrizes \mathbf{A} e \mathbf{B} , uma vez que se pretende obter os ganhos no domínio discreto. Assim, a matriz de realimentação de estados que contém os ganhos discretos é dada por:

$$\mathbf{K}_d = [k_{d1} \ k_{d2} \ k_{d3}] \quad (4.32)$$

$$\mathbf{K}_d = [2.04 \ 0.21 \ 2.79] \quad (4.33)$$

Em seguida são apresentados os resultados de simulação combinada do sistema servo quando é aplicado na sua entrada um sinal de referência, expresso em graus. No algoritmo implementado em MATLAB, o sinal de controlo u é dado pela seguinte equação:

$$u = k_{d1}(r - x_1) - [0 \ k_{d2} \ k_{d3}] \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \quad (4.34)$$

De referir que, na equação anterior, as variáveis de estado posição, velocidade e torque são representadas por x_1 , x_2 e x_3 , respetivamente.

4.5.3.2 Resposta do Sistema a uma entrada de referência bipolar

O resultado que se apresenta de seguida mostra o comportamento do sistema servo para o caso em que o sinal de referência é bipolar variando entre -40 e 40 graus. Apresenta-se a evolução da posição do sistema, bem como as variáveis de estados velocidade e torque e também o sinal de controlo u .

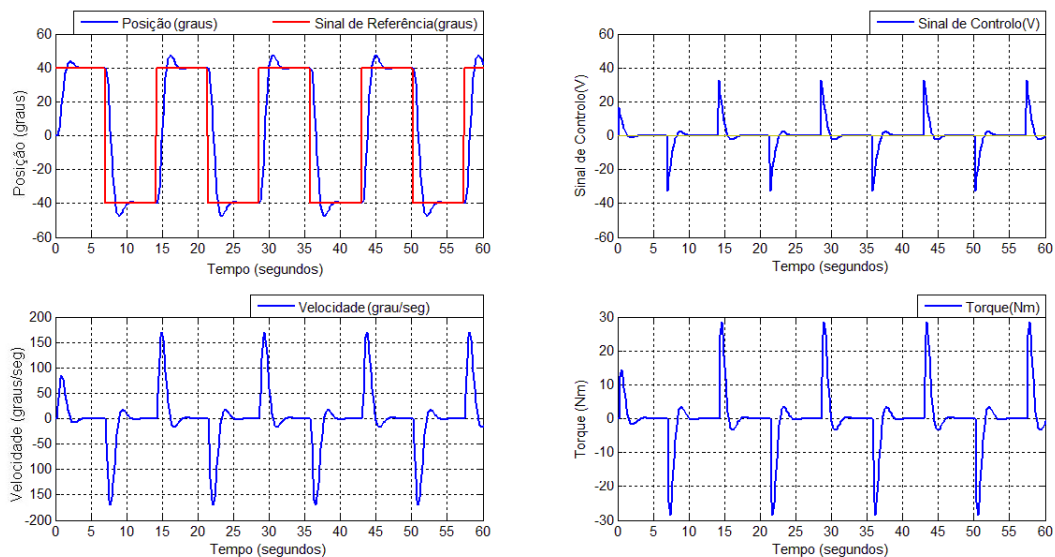


Figura 34 - Resposta do sistema servo em modo combinado para uma referência de [-40,40] graus

De acordo com a figura anterior, utilizou-se o método combinado para simular o comportamento do sistema. Apresenta-se a evolução do sistema em função do tempo (segundos) que depende da relação entre o instante de amostragem k e o período de amostragem h através da expressão $(k * h)$.

O sistema servo apresenta um comportamento em tudo semelhante ao modo contínuo apresentado no tópico 4.5.2.5 no que respeita aos requisitos do sistema para determinação dos pólos da malha fechada, em que o tempo de estabelecimento é cerca de 3 segundos e a sobrelevação é aproximadamente 10%.

Verifica-se também que o sistema atinge o estado estacionário em cerca de 5 segundos. No estado estacionário, a posição do sistema servo corresponde à posição de referência enquanto que a velocidade, torque e sinal de controlo corresponde ao valor zero.

4.5.3.3 Introdução de ruído no sistema

Neste tópico é introduzido ruído no sistema com o objetivo de analisar o comportamento do sistema servo na presença de ruído. O ruído introduzido no sinal de saída do sistema é um ruído branco. Numa implementação real em que se realiza o controlo de um sistema, a utilização de um codificador de quadratura para leitura de posição e/ou velocidade pode gerar com grande facilidade ruído nas medições destes dois parâmetros.

De seguida apresenta-se o diagrama de blocos do sistema de posicionamento com a introdução de ruído branco na saída.

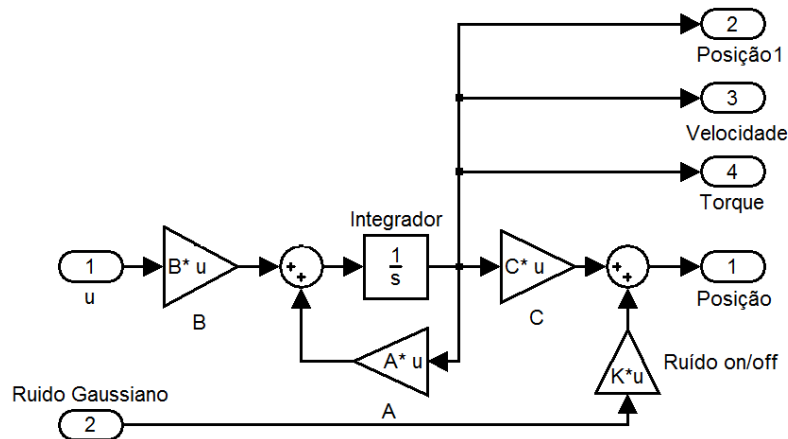


Figura 35 - Diagrama de blocos do sistema a controlar com ruído Gaussiano

Por observação da figura anterior, a entrada do sistema a controlar é dada pelo sinal de entrada u . A saída do sistema que se pretende controlar corresponde à posição, onde é inserido ruído branco gaussiano. São apresentados de seguida os resultados referentes ao comportamento do sistema na presença de ruído gaussiano branco em simulação combinada.

4.5.3.4 Resposta do sistema com influência de ruído

O ruído introduzido no sistema apresenta um desvio padrão de 1 e tem uma média de 0. Os resultados apresentados de seguida mostram o comportamento da posição (graus) do sistema e o erro de simulação, em graus.

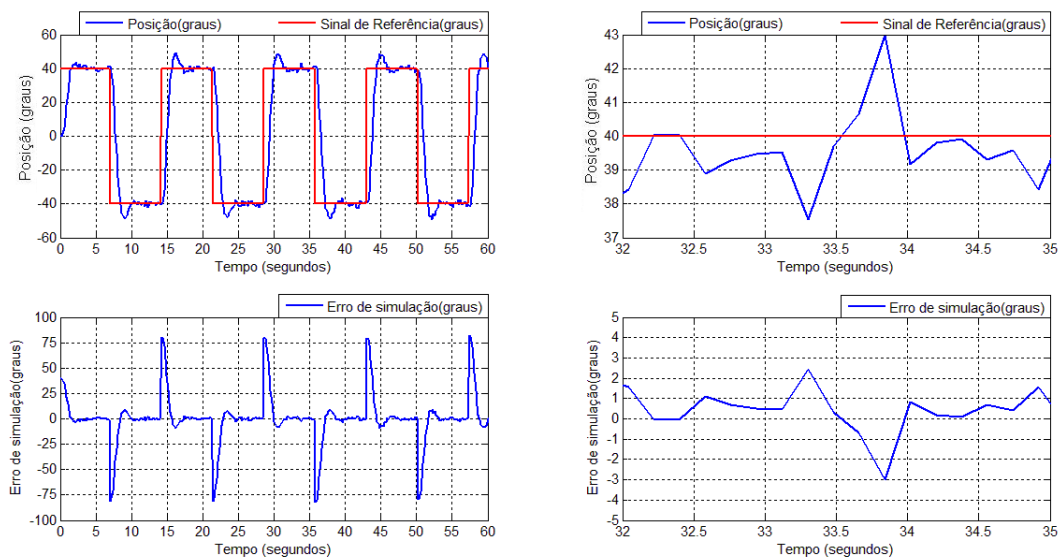


Figura 36 - Influência de ruído branco gaussiano na resposta do sistema servo

Por observação da figura anterior, verifica-se que a posição apresenta ruído em estado estacionário. Por exemplo, no intervalo de tempo que decorre entre $t = 32$ segundos e $t = 35$ segundos, a posição medida na saída do sistema é a mesma da posição de referência. Contudo, esta apresenta ruído, facilmente visível no resultado da Figura 36. O erro entre a posição

medida e a posição de referência no estado estacionário é inferior a 3 graus, estando assim associado a um erro relativo inferior a 8%.

4.5.3.5 Introdução de uma carga externa no sistema

Como visto anteriormente, a presença de ruído nos sensores de medida de posição e/ou velocidade afeta a precisão de medição da saída do sistema. Nesta seção, simula-se o sistema introduzindo uma carga externa. De seguida apresenta-se o diagrama de blocos do sistema a controlar com a introdução de uma carga.

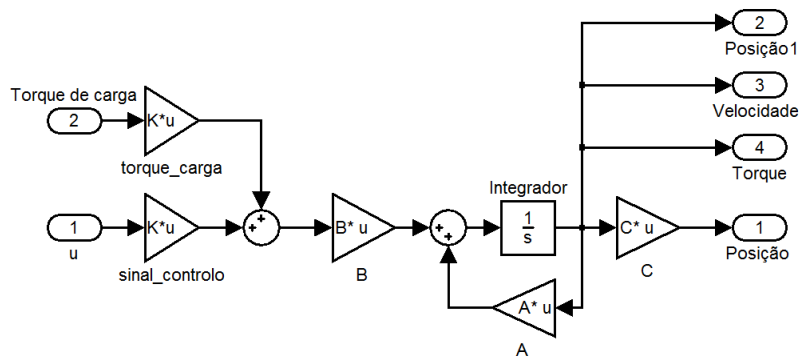


Figura 37 - Introdução de um torque de carga externo à planta do sistema contínuo

Com a introdução de um torque de carga externo à planta, o modelo em espaço de estados da posição é o modelo apresentado no tópico 3.4.2. Este modelo apresenta a matriz B com um segundo termo $-\frac{n}{J_0}$. O torque de carga aplicado ao sistema vai atuar sobre este termo, que é o inverso do momento de inércia da combinação motor DC, sistema de engrenagens e carga.

Na figura anterior, o diagrama do sistema a controlar apresenta duas entradas: o sinal de controlo u e o torque de carga. De seguida, são apresentados os resultados do comportamento do sistema servo quando lhe é aplicado uma carga externa.

4.5.3.6 Influência de uma carga externa na resposta do sistema

Os resultados que se apresentam neste tópico referem-se ao comportamento do sistema servo quando é sujeito a um torque de carga externo de valor fixo e a um torque de carga externo variável. Em primeiro lugar, aplica-se um torque de carga externo de valor fixo 2 N.m. A resposta da posição do sistema servo é apresentada de seguida, bem como o erro entre a posição medida e a posição de referência. Os resultados referem-se à posição do sistema com a introdução de uma carga externa e ao erro de simulação, dado em graus.

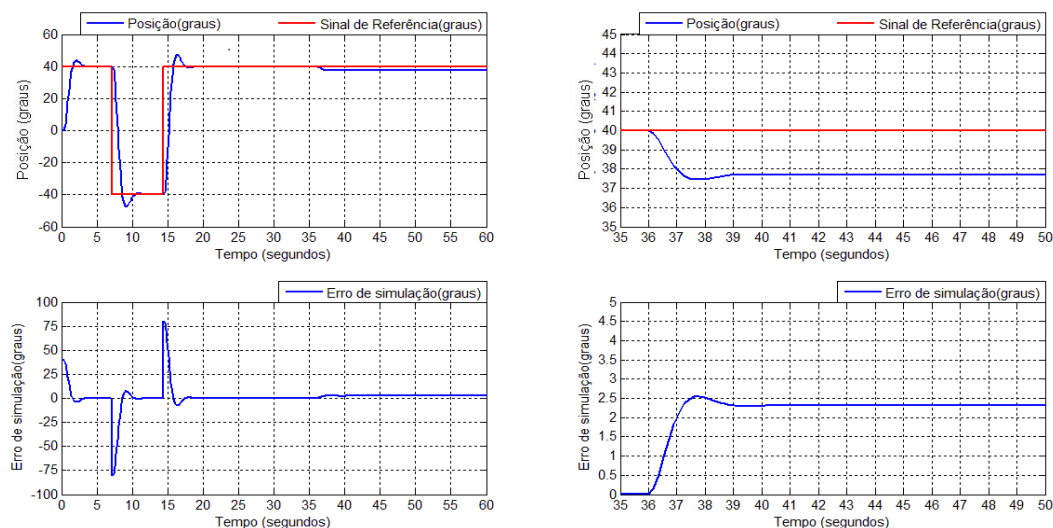


Figura 38 - Influência de um torque de carga externo de 2 N.m na resposta do sistema

O torque de carga externo foi aplicado ao sistema a partir do instante de tempo $t = 36$ segundos. De acordo com a Figura 38, a posição do sistema em estado estacionário é igual à posição de referência. No instante em que é aplicado o torque de carga, a posição sofre uma ligeira alteração no seu valor, ficando assim com um erro em relação à posição de referência de cerca de 2 graus. Assim, é possível concluir que, com um torque de carga de valor constante, o sistema servo apresenta um erro em regime estacionário. Para simulação de um torque de carga variável, é aplicado um sinal sinusoidal com uma amplitude de 2 N.m e período de aproximadamente 3 segundos. Este torque de carga foi aplicado no instante de tempo considerado no caso anterior. Sendo assim, os resultados obtidos são os seguintes.

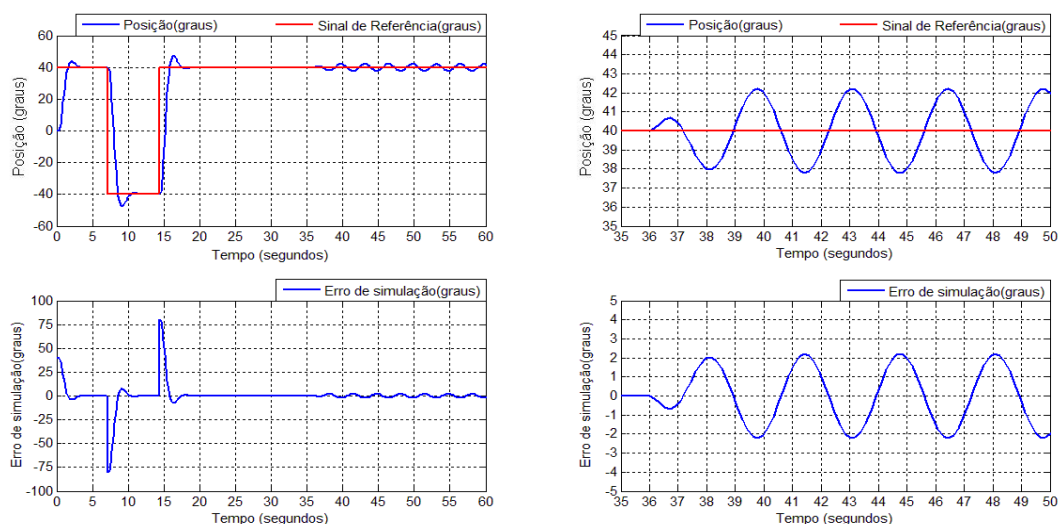


Figura 39 - Influência de um torque de carga externo variável na resposta do sistema

Analisando a figura anterior, o comportamento da posição do sistema servo a partir do instante de tempo $t = 35$ segundos apresenta características sinusoidais. O torque de carga variável aplicado ao sistema no estado estacionário provoca um desvio sinusoidal na sua posição. Este comportamento do sistema é visível na Figura 39. O erro inerente à diferença

entre a posição do sistema servo e a posição de referência é menor que 2.5 graus, correspondendo a uma percentagem de erro inferior a 6%. No entanto, nos resultados obtidos não existe qualquer fator de correção do erro em estado estacionário.

4.5.3.7 Introdução de um ganho de compensação na entrada de referência

Neste tópico simula-se o sistema utilizando um controlo por realimentação total de estados, de acordo com a seguinte diagrama de blocos [16].

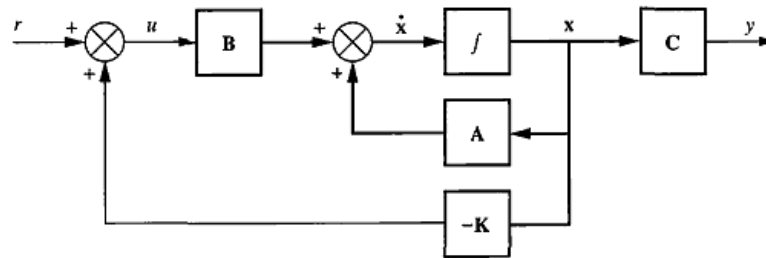


Figura 40 - Diagrama de blocos de um sistema com realimentação total de estados

Para a simulação combinada do sistema da Figura 40, considera-se o diagrama de blocos do sistema a controlar, implementado em Simulink (Figura 27). O novo sinal de controlo u na forma discreta é dado pela seguinte equação.

$$u = r - [k_{d1} \ k_{d2} \ k_{d3}] \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \quad (4.35)$$

De seguida apresenta-se a resposta do sistema para uma entrada de referência bipolar de $[-40,40]$ graus, utilizando os ganhos de realimentação de estados calculados na seção 4.5.3.1.

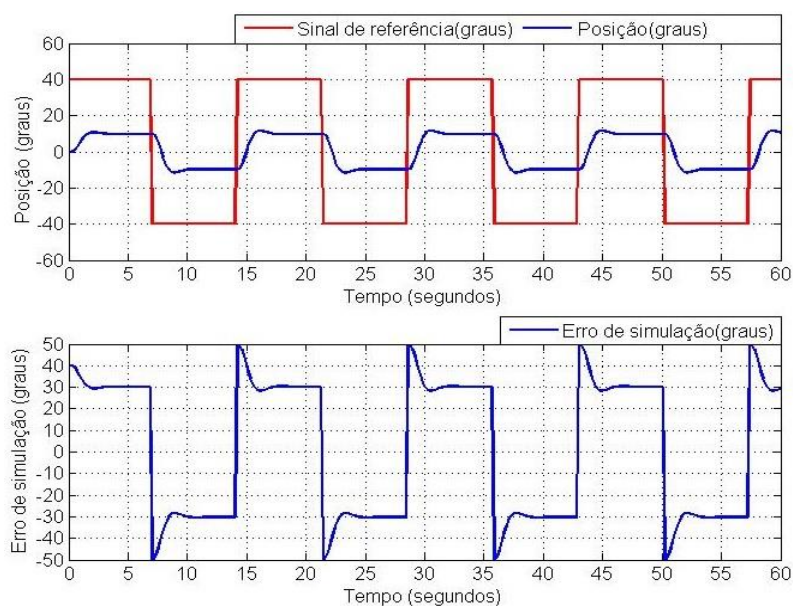


Figura 41 - Resposta do sistema com realimentação total de estados

Analizando de uma forma breve o resultado anterior, o sistema atinge o estado estacionário com um erro considerável. O erro no estado estacionário é cerca de 30 graus, o que permite concluir que o sistema não segue a posição de referência. Para “contornar” este problema do erro no estado estacionário, é introduzido um ganho de pré-compensação na entrada de referência, fora da malha de realimentação de estados. A introdução deste ganho tem o objetivo de corrigir o erro no estado estacionário. No entanto, este ganho é obtido por “tentativa e erro” até que a posição do sistema siga a posição de referência com o menor erro possível. Com isto, o novo sinal de controlo é dado pela seguinte equação.

$$u = r N - [k_{d1} \ k_{d2} \ k_{d3}] \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \quad (4.36)$$

Para um ganho de compensação $N = 1.55$, apresenta-se de seguida a resposta do sistema com a introdução do ganho N na entrada de referência.

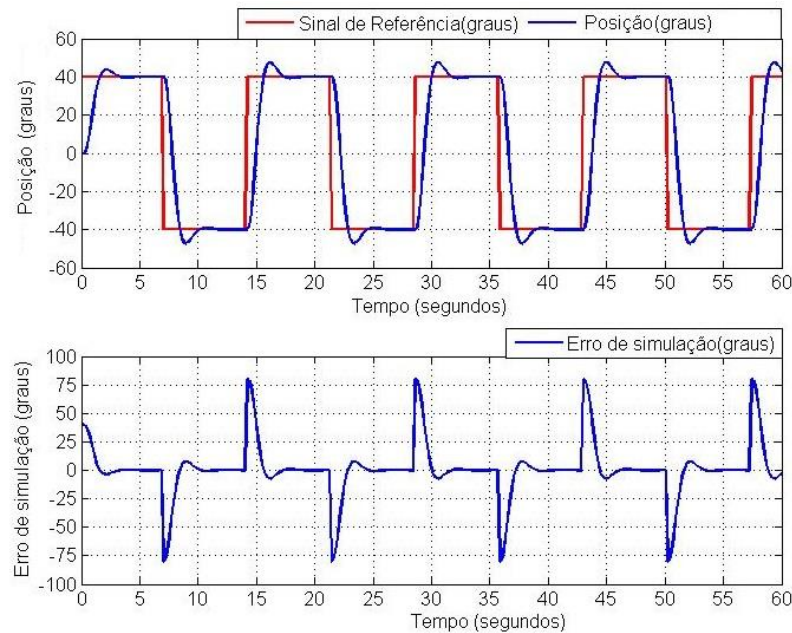


Figura 42 - Resposta do sistema com a introdução de um ganho de pré-compensação

Analizando o resultado da Figura 42, verificou-se uma correção do erro no estado estacionário, sendo que a posição do sistema seguiu a posição de referência. No entanto, esta solução não é prática, uma vez que o ganho N é obtido por um método de “tentativa e erro”.

4.6 Sistema com Ação Integral

No tópico 4.5 foi feita a abordagem ao sistema servo, onde se procedeu à modelação e simulação do mesmo na presença de ruído nos sensores de medição da saída e na presença de uma carga externa. Para corrigir o erro no estado estacionário, utilizou-se um método de introdução de um ganho de pré-compensação, mostrando-se que não era uma solução prática. Para a introdução de ação integral no sistema servo são efetuadas alterações no seu modelo de espaço de estados, de forma a que o erro em estacionário seja corrigido ou atenuado.

Analogamente ao que se realizou no t3pico 4.5, neste t3pico efetuam-se as simula33es do sistema com a33o integral no dom3nio cont3nuo e, posteriormente utiliza-se o m3todo combinado.

4.6.1 Dom3nio Cont3nuo

Neste t3pico 3 implementado o sistema servo com a33o integral. Para isso 3 inserido um integrador no caminho para a frente (*feedforward path*) entre o comparador do erro e a entrada do sistema a controlar. O sistema servo com a33o integral 3 dado pela figura seguinte [19].

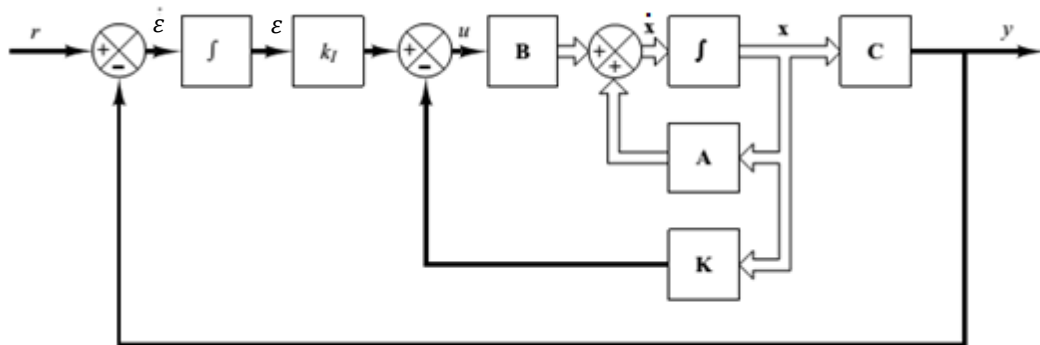


Figura 43 - Sistema Servo com A33o Integral

Por observa33o da Figura 43, o sinal de controlo referente ao sistema servo com a33o integral 3 dado pela seguinte equa33o:

$$u = k_i \varepsilon - Kx \quad (4.37)$$

O ganho do integrador 3 dado por k_i e a sua sa3da 3 representada por ε . A derivada de ε representa o erro (diferen3a entre a refer3ncia e a sa3da) e 3 dada pela seguinte equa33o

$$\dot{\varepsilon} = r - y = r - Cx \quad (4.38)$$

O processo seguinte na modela33o deste sistema consiste na altera33o do seu modelo de espa3o de estados, de forma a incluir o termo do integrador.

4.6.1.1 Modifica33o do modelo de espa3o de estados

Segundo [28], para o sistema com a33o integral, o termo com o integrador que se encontra no caminho para a frente entre o comparador do erro e a entrada do sistema com realimenta33o de estados 3 adicionado ao modelo de espa3o de estados. De seguida, apresenta-se o modelo de espa3o de estados modificado (no dom3nio cont3nuo).

$$\begin{bmatrix} \frac{d\theta(t)}{dt} \\ \frac{d\omega(t)}{dt} \\ \frac{d\tau(t)}{dt} \\ \frac{d\varepsilon(t)}{dt} \end{bmatrix} = \mathbf{A}_i \begin{bmatrix} \theta(t) \\ \omega(t) \\ \tau(t) \\ \varepsilon(t) \end{bmatrix} + \mathbf{B}_i u + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} r \quad (4.39)$$

$$\mathbf{A}_i = \begin{bmatrix} \mathbf{A} & \mathbf{0} \\ -\mathbf{C} & 0 \end{bmatrix} \quad (4.40)$$

$$\mathbf{B}_i = \begin{bmatrix} \mathbf{B} \\ 0 \end{bmatrix} \quad (4.41)$$

$$\mathbf{C}_i = [n \ 0 \ 0 \ 0] \quad (4.42)$$

Com a introdução da ação integral no sistema, o modelo do sistema na forma de espaço de estados foi aumentado, tendo sido adicionado mais uma equação de estado. De seguida, são calculados os novos pólos deste sistema com ação integral, pólos esses que são utilizados no cálculo dos ganhos de realimentação de estados.

4.6.1.2 Determinação da matriz de Realimentação de estados do sistema aumentado

Conforme referido na seção anterior, o sistema com ação integral apresenta um pólo adicional comparativamente ao sistema servo da Figura 29, ficando assim um sistema com 4 pólos. O pólo adicional está associado ao integrador adicional que se encontra no caminho para a frente (*feedforward path*) da Figura 43. Considera-se o modelo do sistema em malha aberta apresentado no tópico 4.5.2.2. Utilizando os mesmos requisitos do sistema para a malha fechada, ou seja, um tempo de estabelecimento de 3 segundos e um valor percentual de pico de 10 %, apresenta-se de seguida os pólos de malha fechada. Tal como efetuado anteriormente, deve-se posicionar o terceiro pólo o mais afastado possível da componente real dos pólos dominantes. Assim, o terceiro pólo é posicionado a três vezes da componente real dos pólos dominantes e o pólo adicional (pólo da ação integral) a cinco vezes da componente real. De seguida apresenta-se a matriz com os pólos de malha fechada.

$$\mathbf{p}_{ci} = [p_{ci1} \ p_{ci2} \ p_{ci3} \ p_{ci4}] \quad (4.43)$$

$$\mathbf{p}_{ci} = [-1.3 + 1.81i \ -1.3 - 1.81i \ -4 \ -6.6] \quad (4.44)$$

Os pólos do sistema com ação integral em modo contínuo são representados pela matriz \mathbf{p}_{ci} . De acordo com [19], a matriz de realimentação de estados com o ganho da ação integral é dada pela seguinte equação:

$$\mathbf{K}_{ci} = [\mathbf{K} \ | \ -k_i] \quad (4.45)$$

$$K = [9.4 \ 2.18 \ 11.38] \quad (4.46)$$

A matriz de ganhos apresentada anteriormente na equação 4.46 resulta da utilização do método de *Ackermann*, em que o ganho $k_i = 20.34$ corresponde ao ganho da ação integral.

4.6.1.3 Construção do sistema servo com ação integral em Simulink

Recorrendo à ferramenta Simulink, o diagrama de blocos do sistema com ação integral correspondente à Figura 43 é apresentado de seguida.

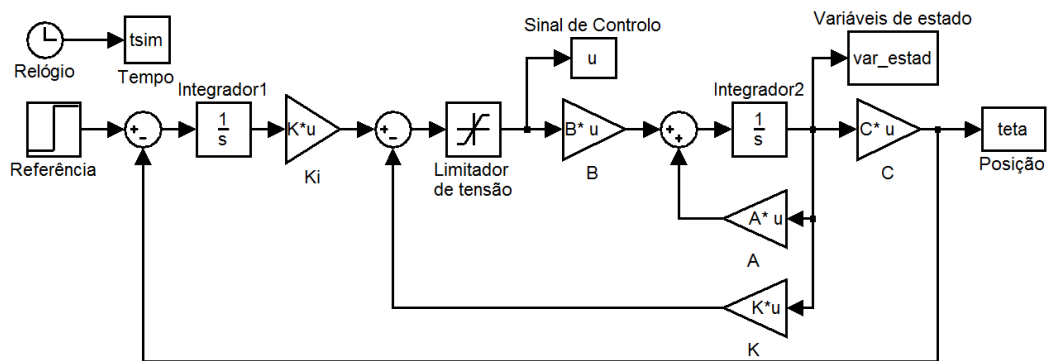


Figura 44 - Diagrama de blocos em Simulink do sistema servo com ação integral

O diagrama de blocos da figura anterior representa o sistema servo em modo contínuo com ação integral. De referir que o bloco representado por K representa a matriz com os ganhos de realimentação de estados K e o bloco K_i corresponde ao ganho do termo integral adicionado ao sistema. De seguida são apresentados os resultados com o controlador de posição por realimentação de estados e com ação integral.

4.6.1.4 Resposta do Sistema com Ação Integral

Para simular a resposta do sistema servo em modo contínuo considerou-se uma posição de referência de 40 graus e um tempo de simulação de 60 segundos. De acordo com o tópico anterior, é simulada a resposta do sistema tendo em consideração que a resposta transitória apresenta uma sobrelevação de 10% e um tempo de estabelecimento de 3 segundos. De seguida é apresentada a evolução da posição do sistema, das variáveis de estado velocidade e torque e, por fim, do sinal de controlo u .

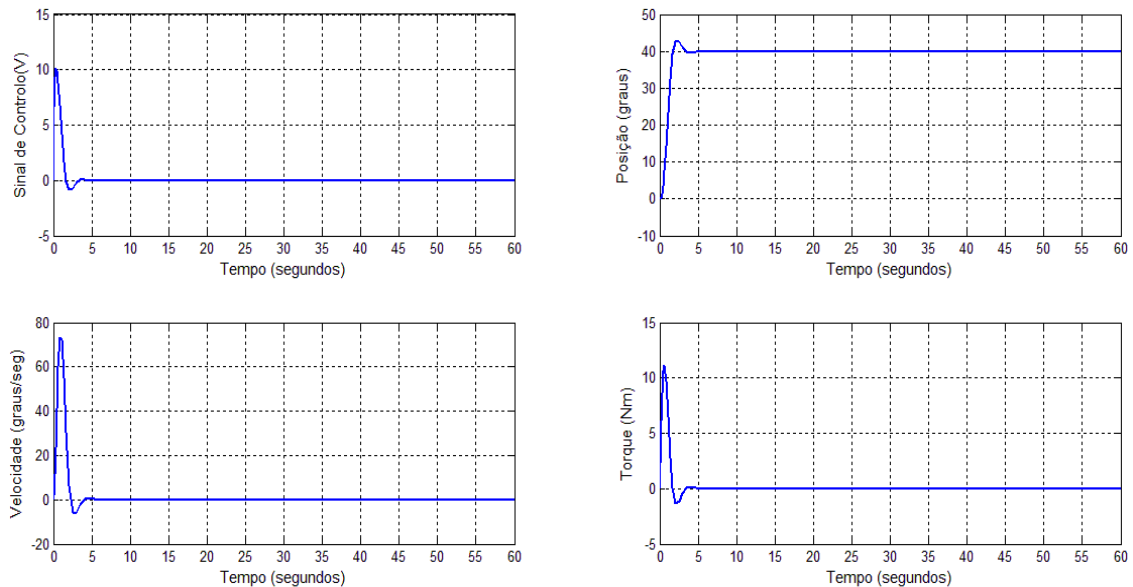


Figura 45 - Resposta do Sistema Servo com Ação Integral para uma referência de 40 graus

Por observação da figura anterior, é possível concluir que o sistema com ação integral apresenta um comportamento semelhante ao do sistema servo sem ação integral, abordado no tópico 4.5. De referir que os resultados apresentados neste tópico referem-se a um sistema que não apresenta qualquer ruído de medição das variáveis de estado nem é sujeito a um torque de carga.

De seguida é feita a análise a este sistema em modo combinado com a introdução de ruído e um torque de carga.

4.6.2 Simulação Combinada

Para o sistema apresentado na Figura 43, o termo integral permite ao sistema reagir a certos fatores como o ruído existente nos sensores de medida e/ou a variações de carga externas ao sistema. Nesta seção é analisado o comportamento deste sistema na presença de ruído e variação de carga. A metodologia utilizada para a simulação combinada do sistema com ação integral é a mesma do tópico 4.5.3.

4.6.2.1 Discretização das Matrizes de Ganho

Para implementação do controlador discreto em espaço de estados, é necessário obter a matriz de ganho na forma discreta. A obtenção desta matriz foi feita utilizando a mesma metodologia do tópico 4.5.3.1.

Para a determinação dos pólos de malha fechada no domínio contínuo, consideram-se os mesmos requisitos utilizados no tópico 4.6.1.2. Como já referido, o terceiro pólo é posicionado a três vezes da componente real dos pólos dominantes e o quarto pólo (pólo da ação integral) a cinco vezes da componente real, ficando assim localizados em -4 e -6.66, respetivamente. Considerando um tempo de amostragem de 180ms, os pólos do sistema no plano Z obtidos pela equação 4.29 são os seguintes:

$$\mathbf{p}_{zi} = [p_{zi1} \ p_{zi2} \ p_{zi3} \ p_{zi4}] \quad (4.47)$$

$$\mathbf{p}_{zi} = [0.74 + 0.25i \ 0.74 - 0.25i \ 0.48 \ 0.3] \quad (4.48)$$

Uma vez que estes pólos no plano Z se encontram localizados dentro do círculo unitário, o sistema cumpre os requisitos de estabilidade. Recorrendo ao método de *Ackermann*, as matrizes de ganho de realimentação na sua forma discreta são [19]:

$$\mathbf{K}_d = [\mathbf{K}_{di} \ | \ -k_{dint}] \quad (4.49)$$

$$\mathbf{K}_{di} = [6.01 \ 1.37 \ 6.78] \quad (4.50)$$

O ganho adicional referente ao termo integral é representado por k_{dint} e tem o valor de $k_{dint} = -11.65$. A matriz \mathbf{K}_d contém os ganhos discretos obtidos pelo método de *Ackermann*.

Substituindo na expressão 4.49 os valores dos ganhos obtidos pelo método de *Ackermann*,

$$\mathbf{K}_d = [6.01 \ 1.37 \ 6.78 \ | \ 11.65] \quad (4.51)$$

4.6.2.1 Obtenção do sinal de controlo do sistema servo com ação integral

Considerando o diagrama de blocos da Figura 35, o sinal de controlo u referente ao sistema com ação integral na sua forma discreta é dada pela seguinte equação:

$$u(k) = k_{dint}\varepsilon(k) - \mathbf{K}_{di}\mathbf{x}(k) \quad (4.52)$$

De acordo com equação anterior, a saída do termo integral $\xi(k)$ é dada por:

$$\varepsilon(k) = \varepsilon(k-1) + (r(k) - y(k)) \quad (4.53)$$

Substituindo a equação 4.53 na equação 4.52, tem-se

$$u(k) = k_{dint}(\varepsilon(k-1) + (r(k) - y(k))) - \mathbf{K}_{di}\mathbf{x}(k) \quad (4.54)$$

A equação anterior representa o sinal de controlo u dependente do instante de amostragem k . Ainda na mesma equação, $\varepsilon(k-1)$ representa o integral do erro no instante de amostragem anterior, $r(k)$ o sinal de referência, $y(k)$ o sinal de saída ou posição e $\mathbf{x}(k)$ as variáveis de estado. Como já visto anteriormente, o sinal $y(k)$ aparece nos resultados simulados afetado do termo n , que corresponde à relação das engrenagens do sistema. De forma análoga ao sistema servo sem ação integral, apresentam-se os resultados de simulação do sistema com ação integral na presença de ruído e com introdução de uma carga externa.

4.6.2.2 Influência de ruído na resposta do sistema servo com ação integral

O ruído introduzido na saída do sistema servo com ação integral é um ruído gaussiano branco com as mesmas características do ruído utilizado anteriormente no sistema servo sem ação integral (tópico 4.5.3.2).

Com esta simulação pretende-se a comparação com os resultados obtidos no caso do sistema servo sem ação integral, em que se introduziu um sinal de ruído gaussiano branco com um desvio padrão de 1 e uma média de 0.

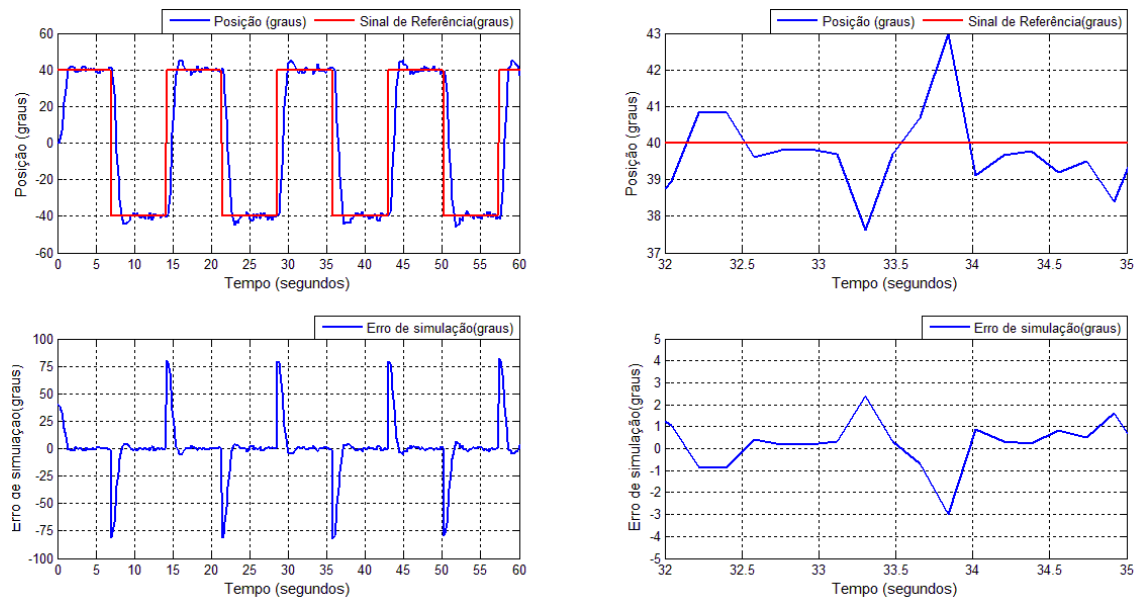


Figura 46 - Influência de ruído branco gaussiano na resposta do sistema com ação integral

A figura anterior apresenta a evolução da posição, em graus, e o erro que se obtém entre a posição medida e a posição de referência, apresentado em graus.

Considerando por exemplo o intervalo de tempo de $t = [32, 35]$ segundos, intervalo em que o sistema se encontra em estado estacionário, observa-se a existência de ruído na medição da posição. No entanto, o erro entre a posição medida e a posição de referência neste intervalo de tempo é inferior a 3 graus, correspondendo assim a um erro relativo inferior a 8%. Comparativamente ao resultado obtido para o caso do sistema servo sem ação integral, não se verificou melhoria do erro em estado estacionário proveniente da medição da saída do sistema. Conclui-se que a introdução de ação integral neste sistema não permite corrigir erros de medição da saída (em sistemas reais podem ser considerados ruídos de medição de sensores).

4.6.2.3 Influência de uma carga externa na resposta do sistema com ação integral

Neste tópico apresenta-se os resultados de simulação de uma carga que é aplicada externamente ao sistema com ação integral. Consideram-se as mesmas condições de simulação do tópico 4.5.3.4, onde se aplicaram dois tipos de torque de carga: um torque de carga de valor fixo e outro de valor variável. Com esta simulação pretende-se comparar o sistema com ação integral com o sistema servo sem ação integral no que respeita ao comportamento do erro em estado estacionário.

Considera-se o diagrama de blocos da Figura 37 que, interligado com o algoritmo de controlo implementado em MATLAB, estabelecem o modo combinado. Utiliza-se o sinal de controlo determinado pela equação 4.54 que inclui a ação integral. Numa primeira fase é aplicado um torque de carga de valor 2 N.m, valor este que se mantém fixo durante todo o processo de amostragem. De seguida são apresentados os resultados com a evolução da posição do sistema servo com ação integral e o erro entre a posição medida e a sua posição de referência perante um torque de carga de valor referido anteriormente.

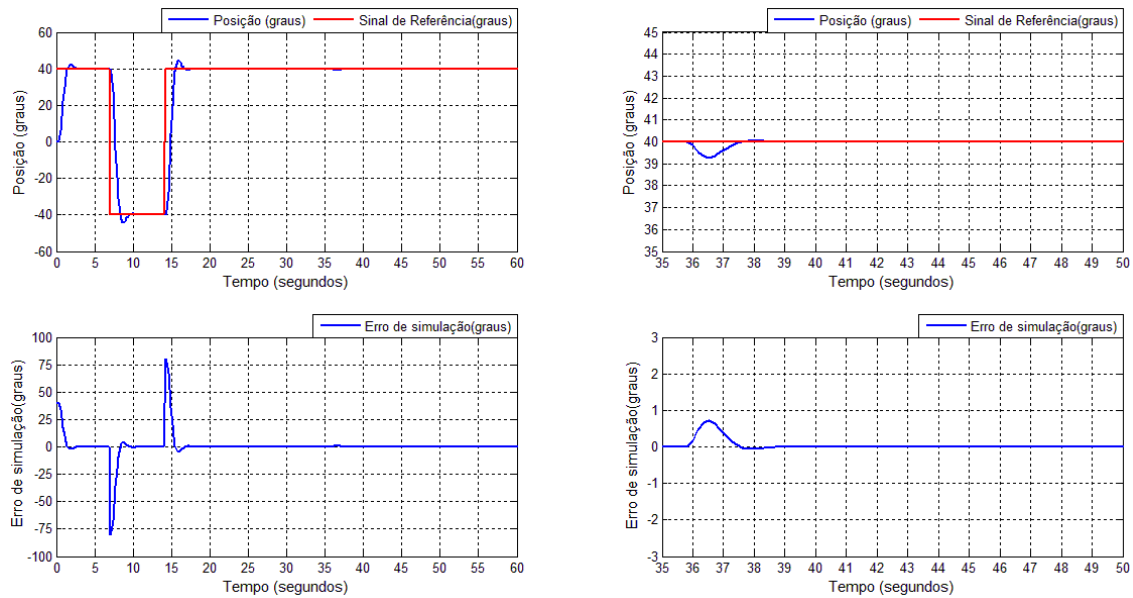


Figura 47 - Influência de um torque de carga fixo na resposta do sistema com ação integral

Por observação da Figura 47, foi aplicado um torque de carga no instante de tempo $t=36$ segundos. A posição do sistema diminui cerca de 0.75 graus em relação à posição de referência, demorando cerca de 3 segundos a atingir de novo a posição de referência. O erro entre a posição de referência e a posição do sistema é inferior a 1 grau, no instante em que é aplicado o torque de carga. Comparativamente ao resultado obtido no caso do sistema servo sem ação integral, o erro entre a posição de referência e a posição do sistema é corrigido. Com isto, conclui-se que a ação integral no sistema permite corrigir o erro em estado estacionário para o caso de um torque de carga de valor fixo.

De seguida apresenta-se o resultado referente ao comportamento do sistema com ação integral quando se introduz uma carga variável com características sinusoidais. Considera-se as mesmas condições de simulação do tópico 4.5.3.4.

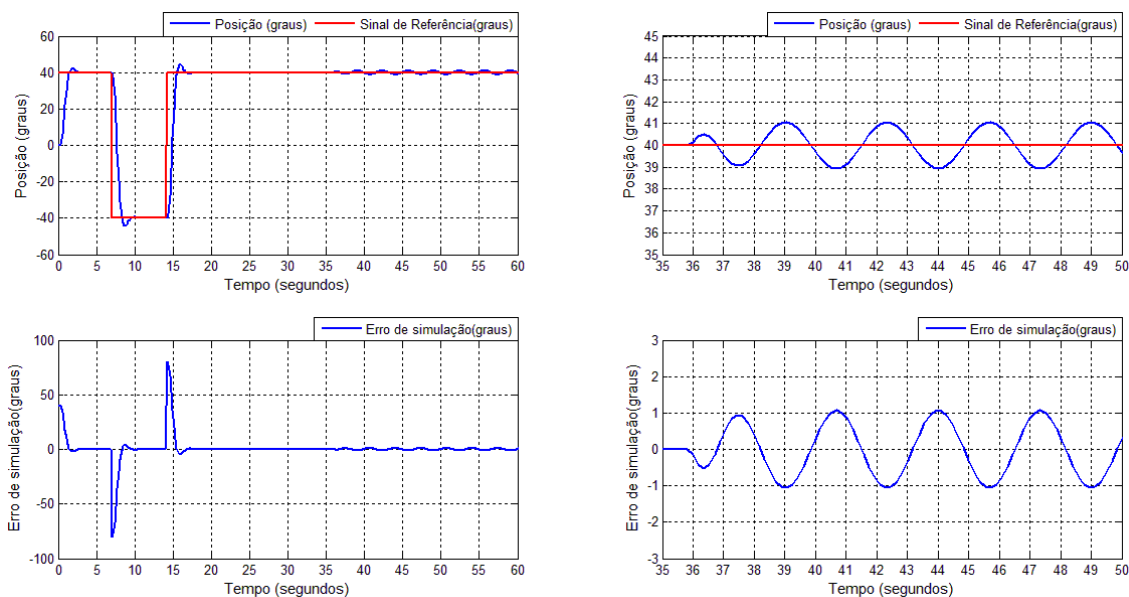


Figura 48 - Influência de um torque de carga variável na resposta do sistema com ação integral

O resultado apresentado anteriormente mostra que, a partir do instante de tempo $t = 36$ segundos é aplicado um torque de carga de valor variável com as mesmas características sinusoidais da carga utilizada no tópico 4.5.3.4. A posição do sistema apresenta um comportamento sinusoidal em torno da posição de referência. O erro entre a posição medida e a posição de referência é cerca de 1 grau.

Comparativamente aos resultados do tópico 4.5.3.4, o erro no estado estacionário do sistema com ação integral é menor que o erro obtido para o caso do sistema sem ação integral. Pode-se então concluir que a ação integral é um fator de correção de erro em estado estacionário.

4.7 Observador de Estados

Nas seções anteriores efetuaram-se simulações do controle do sistema utilizando realimentação de estados com e sem ação integral. Para isso, considerou-se o fato das variáveis de estado serem mensuráveis, não havendo necessidade de as estimar. No entanto, caso não seja possível medir as variáveis do sistema, estas podem ser estimadas utilizando observadores de estado. Existem dois tipos de observadores: observadores de ordem completa e de ordem reduzida. Neste tópico é feito o dimensionamento de um observador de estados de ordem completa, onde é efetuada a estimação de todas as variáveis de estado e, posteriormente, a simulação do sistema.

4.7.1 Diagrama de blocos do sistema com Observador de estados

O diagrama de blocos apresentado neste tópico refere-se ao sistema a controlar com realimentação de estados utilizando um observador de estados de ordem completa [19]. Trata-se de um sistema servo com uma referência na entrada e não possui ação integral. Relativamente ao observador de estados, \hat{x} representa o vetor de estado estimado e L representa a matriz de ganhos do observador.

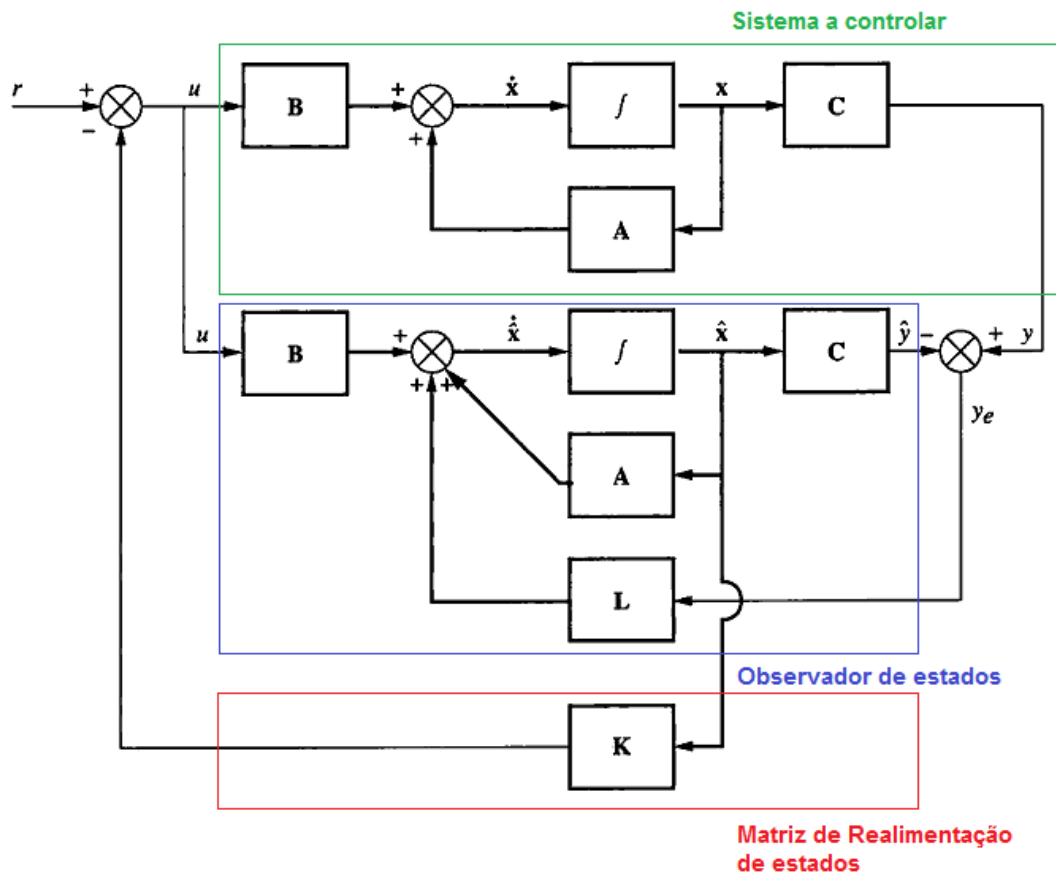


Figura 49 - Diagrama de blocos do sistema servo com observador de estados

4.7.2 Estimativa do vetor de estado

Como já referido anteriormente um observador de estado pode ser de ordem completa ou de ordem reduzida, dependendo se todas as variáveis de estado são mensuráveis ou não. Para o projeto do controlador com realimentação de estados utilizando um observador de estados de ordem completa, utiliza-se uma metodologia semelhante à que foi apresentada no tópico 4.5.2. Numa primeira fase verifica-se se o sistema é observável e, posteriormente determina-se as matrizes de ganho L e K . Relativamente à matriz de realimentação de estados, o método de obtenção foi apresentado anteriormente.

De acordo com o observador de estados da Figura 49, é possível obter a estimativa do vetor de estado utilizando a seguinte expressão, na forma discreta:

$$\hat{\mathbf{x}}(k+1) = \mathbf{G}\hat{\mathbf{x}}(k) + \mathbf{H}u(k) + \mathbf{L}y_e(k) \quad (4.55)$$

Na expressão anterior, as matrizes \mathbf{G} e \mathbf{H} correspondem à forma discreta das matrizes \mathbf{A} e \mathbf{B} , respetivamente. Simplificando a expressão 4.54, obtém-se:

$$\hat{\mathbf{x}}(k+1) = (\mathbf{G} - \mathbf{L}\mathbf{C})\hat{\mathbf{x}}(k) + \mathbf{H}u(k) + \mathbf{L}y(k) \quad (4.56)$$

A expressão 4.56 pode ser re-escrita da seguinte forma de modo a simplificar a sua implementação e considerando k o instante atual.

$$\hat{\mathbf{x}}(k) = (\mathbf{G} - \mathbf{LC})\hat{\mathbf{x}}(k-1) + \mathbf{H}u(k-1) + \mathbf{L}y(k-1) \quad (4.57)$$

4.7.2.1 Observabilidade

Um sistema é observável, se a partir das medidas da saída y é possível determinar o estado do sistema. De uma outra forma, diz-se que o sistema é observável se para qualquer estado inicial x_0 , existir um intervalo de tempo finito tal que x_0 pode ser determinado a partir do conhecimento de y . Se o vetor de saída y do sistema for afetado por todas as variáveis de estado então o sistema diz-se completamente observável. Uma regra prática (apresentada pela expressão 4.58) para analisar a observabilidade do sistema consiste em calcular a característica da matriz de observabilidade e verificar se é igual ao número de estados n do sistema. Se isto se verificar, então o sistema é completamente observável.

$$\text{caraterística_matricial}(\mathbf{W}) = n \quad (4.58)$$

A matriz \mathbf{W} representa a matriz de Observabilidade e é dada pela seguinte expressão:

$$\mathbf{W} = \begin{bmatrix} \mathbf{C} \\ \mathbf{CA} \\ \mathbf{CA}^2 \\ \vdots \\ \mathbf{CA}^{n-1} \end{bmatrix} \quad (4.59)$$

4.7.2.2 Método de obtenção da matriz de ganhos \mathbf{L}

Tal como efetuado no cálculo da matriz \mathbf{K} (matriz de realimentação de estados), no caso do observador de estados utiliza-se o método de *Ackermann* para determinar a matriz \mathbf{L} , ainda que com algumas diferenças. Utilizando então o método referido, a matriz de ganhos do observador de estados é dada pela expressão [19]:

$$\mathbf{L} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 1 \end{bmatrix} \mathbf{W}^{-1} q(\mathbf{A}) \quad (4.60)$$

Na expressão 4.60, \mathbf{W}^{-1} representa a inversa da matriz de observabilidade e $q(\mathbf{A})$ a equação característica com base nos pólos selecionados para o observador de estados. A equação característica $q(\mathbf{A})$ foi apresentada anteriormente na expressão 4.22. Na seleção dos pólos do observador, tem que se ter em conta a dinâmica do observador, que terá que ser mais rápida que a dinâmica do controlador do sistema por forma a acompanhar as variações de estado do sistema real. Regra geral, os pólos do observador devem ser 2 a 5 vezes mais rápidos que os pólos do controlador do sistema [19]. Na localização dos pólos no semi-plano esquerdo do plano s , os pólos do controlador do sistema devem estar localizados à direita dos pólos do observador [19]. A expressão 4.60 permite obter a matriz de ganhos do observador de estados

no domínio contínuo. Por outro lado, a matriz de ganhos do observador no domínio discreto é dada na forma [22]:

$$\mathbf{L}_d = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 1 \end{bmatrix} \mathbf{W}_d^{-1} q(\mathbf{G}) \quad (4.61)$$

em que \mathbf{W}_d^{-1} representa a matriz inversa de observabilidade no domínio discreto e $q(\mathbf{G})$ representa a equação característica do observador com base nos pólos discretos selecionados para o observador e na matriz discreta \mathbf{G} . De seguida, apresentam-se as expressões que definem \mathbf{W}_d e $q(\mathbf{G})$.

$$\mathbf{W}_d = \begin{bmatrix} \mathbf{C} \\ \mathbf{C}\mathbf{G} \\ \mathbf{C}\mathbf{G}^2 \\ \vdots \\ \mathbf{C}\mathbf{G}^{n-1} \end{bmatrix} \quad (4.62)$$

$$q(\mathbf{G}) = \mathbf{G}^n + \alpha_1 \mathbf{G}^{n-1} + \dots + \alpha_n \quad (4.63)$$

4.7.3 Simulação Combinada

Para testar a qualidade do observador de estados apresentado anteriormente, é utilizado um controlador de posição do sistema servo com realimentação de estados e um observador de estados (Figura 49). A este sistema é adicionada ação integral, já utilizada em 4.6. De seguida são calculados os ganhos referentes ao observador a partir da seleção dos pólos para a malha fechada e, posteriormente, apresentam-se os resultados da simulação efetuada com o observador de estados.

4.7.3.1 Determinação dos ganhos discretos do Observador

Numa fase inicial, analisou-se a observabilidade do sistema pela característica matricial (*rank*). Utilizando as funcionalidades do MATLAB, verificou-se que o *rank* da matriz de observabilidade corresponde a 3, que é igual ao número de variáveis de estado do sistema. Sendo assim, o sistema é completamente observável.

Uma vez que a simulação é feita em modo combinado, utiliza-se o modelo do sistema a controlar (em Simulink) apresentado na Figura 27 e o controlador discreto (em MATLAB) adotando a mesma metodologia apresentada na seção 4.6.2. Com isto, torna-se apenas necessário determinar a matriz de ganhos na forma discreta do observador \mathbf{L}_d .

No processo de determinação da matriz de ganhos \mathbf{L}_d considerou-se o modelo na forma de espaço de estados do sistema servo sem ação integral definido na seção 4.4.1. De maneira a tornar a dinâmica do observador superior à do controlador do sistema, utilizaram-se os pólos de malha fechada do sistema multiplicados por um ganho de 3. Sendo assim, os pólos do observador encontram-se localizados no plano *s* em $-4 \pm 5.45j$ e -12 . Os pólos discretos encontram-se localizados no plano *Z* em $-0.27 \pm 0.4j$ e 0.11 . Por fim, aplicando o método de *Ackermann* obtém-se a matriz de ganhos do observador:

$$L_d = [2.79 \quad 11.7 \quad 0.77] \quad (4.64)$$

A matriz de ganhos do observador apresentada anteriormente é aplicada na expressão de estimativa do vetor de estado do sistema (expressão 4.57). Relativamente às matrizes G e H , consideram-se as matrizes apresentadas no tópico 4.4.2.1. O sinal que permite o controlo discreto do sistema com observador de estados e ação integral é dado por,

$$u(k) = k_{dint}(\varepsilon(k-1) + (r(k) - y(k))) - K_{di}\hat{x}(k) \quad (4.65)$$

A expressão 4.65 refere-se ao novo sinal de controlo do sistema. De referir que no sinal apresentado as variáveis de estado são estimadas pelo vetor $\hat{x}(k)$ ao contrário do que se verifica na expressão 4.52 em que as variáveis de estado se encontram acessíveis através do vetor $x(k)$. Por fim, utilizam-se os ganhos k_{dint} e K_{di} calculados anteriormente na seção 4.6.2.1.

4.7.3.2 Resposta do sistema a uma referência de posição

Para simular a resposta do sistema a controlar e, posteriormente, analisar o seu comportamento, considerou-se um tempo de simulação de 60 segundos e uma posição de referência (*setpoint*) bipolar que varia entre -40 e 40 graus. As variáveis de estado que definem o sistema são a posição, a velocidade e o torque. Os resultados apresentados de seguida mostram o comportamento das variáveis de estado estimadas e do sinal de controlo através da utilização do observador de ordem completa.

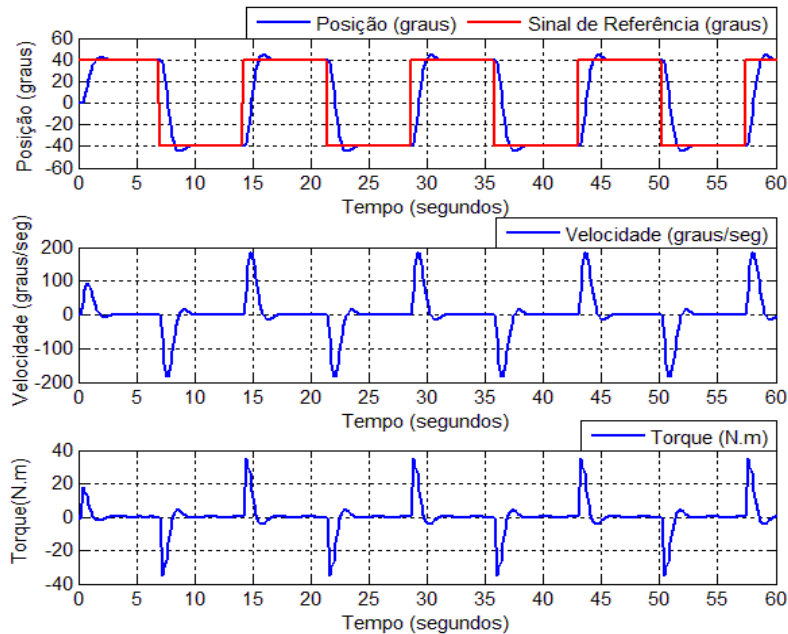


Figura 50 - Comportamento das variáveis estimadas pelo Observador de estados

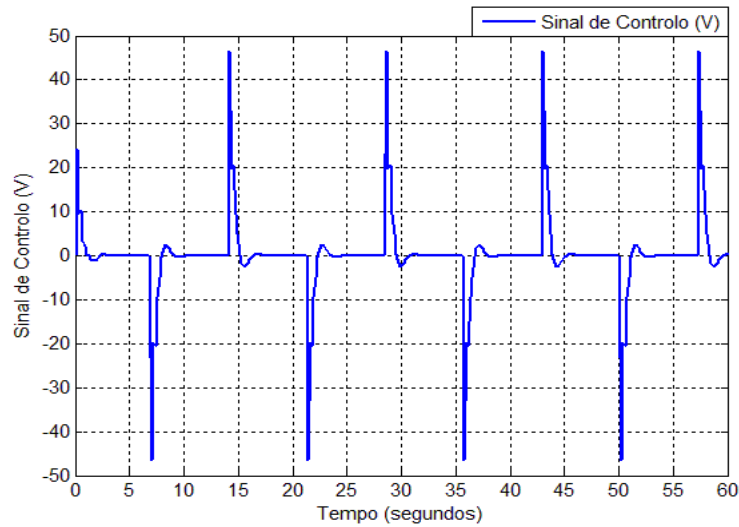


Figura 51 - Evolução do sinal de controle u do sistema com observador de estados

A Figura 50 mostra a evolução dos sinais das variáveis de estado estimadas pelo observador sendo o objetivo o controle de posição do sistema. Analisando a resposta transitória da posição do sistema, apresenta um valor de sobrelevação aproximadamente de 10% e um tempo de estabelecimento inferior a 3 segundos. O sistema demora aproximadamente 3 segundos a atingir o regime estacionário, estabilizando assim na posição de referência. As variáveis de estado velocidade e torque apresentam um pico inicial quando o sistema se encontra no estado transitório. Quando o sistema se aproxima do estado estacionário, as variáveis de estado tendem para o seu valor nulo. O sinal de controle, dado em volts, encontra-se representado na Figura 51.

De seguida são apresentados os sinais das variáveis de estado acessíveis pelo sistema de forma a comparar com os sinais estimados pelo observador de estados mostrados anteriormente.

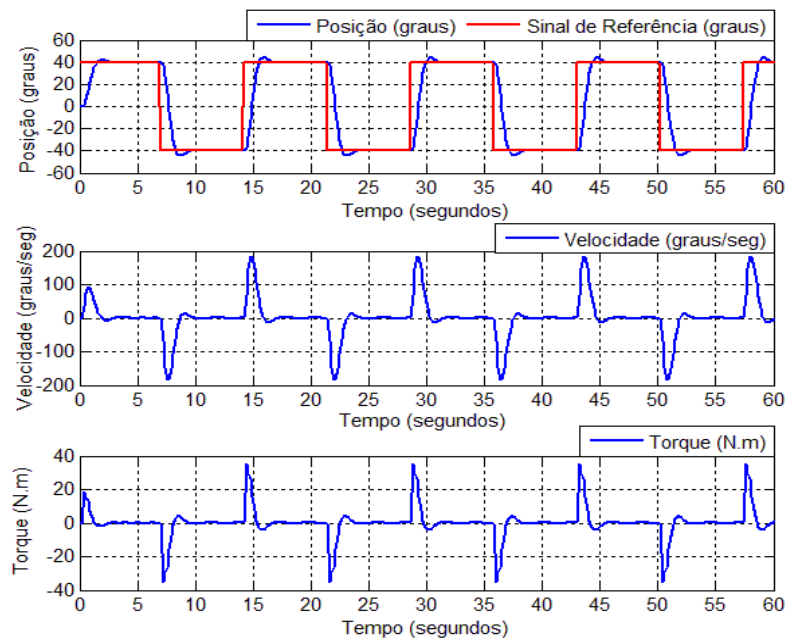


Figura 52 - Comportamento das variáveis de estado medidas pelo sistema

As figuras seguintes mostram a comparação entre os sinais das variáveis de estado estimadas pelo observador e as variáveis de estado medidas pelo sistema. Para analisar as diferenças entre o observador de estados e o controlador do sistema, consideraram-se os intervalos de tempo $t_1 = [0,6]$ segundos e $t_2 = [14,20]$ segundos.

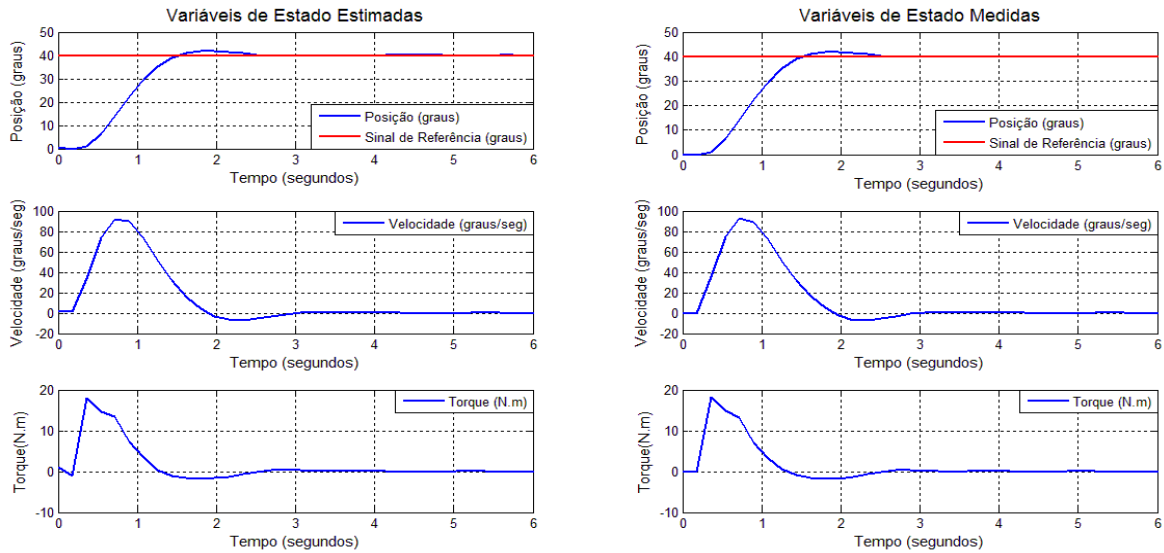


Figura 53 - Comportamento das variáveis de estado estimadas e medidas (intervalo t_1)

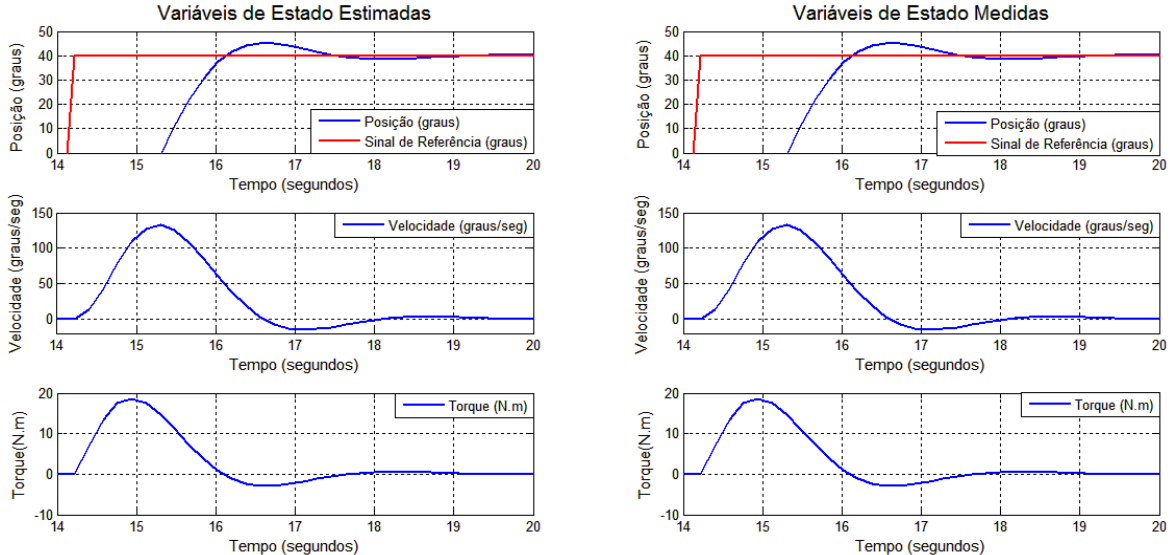


Figura 54 - Comportamento das variáveis de estado estimadas e medidas (intervalo t_2)

Por observação das figuras anteriores, os sinais das variáveis de estado estimadas pelo observador são semelhantes aos sinais das variáveis medidas (ou acessíveis). Analisando os resultados nos dois intervalos de tempo (t_1 e t_2), o erro de estimação por parte do observador é praticamente nulo, podendo assim concluir que o observador de estados foi bem projetado permitindo uma boa estimativa das variáveis de estado posição, velocidade e torque.

Capítulo 5 Plataforma de Ensaios

No presente capítulo, apresenta-se a plataforma de ensaios utilizada no contexto deste trabalho. A plataforma de ensaios inclui a estrutura mecânica, eletrônica de medição e atuação do sistema, fonte de alimentação e cablagem de suporte. Numa fase posterior é descrito o *firmware/software* desenvolvido para a Identificação e Controlo do sistema.

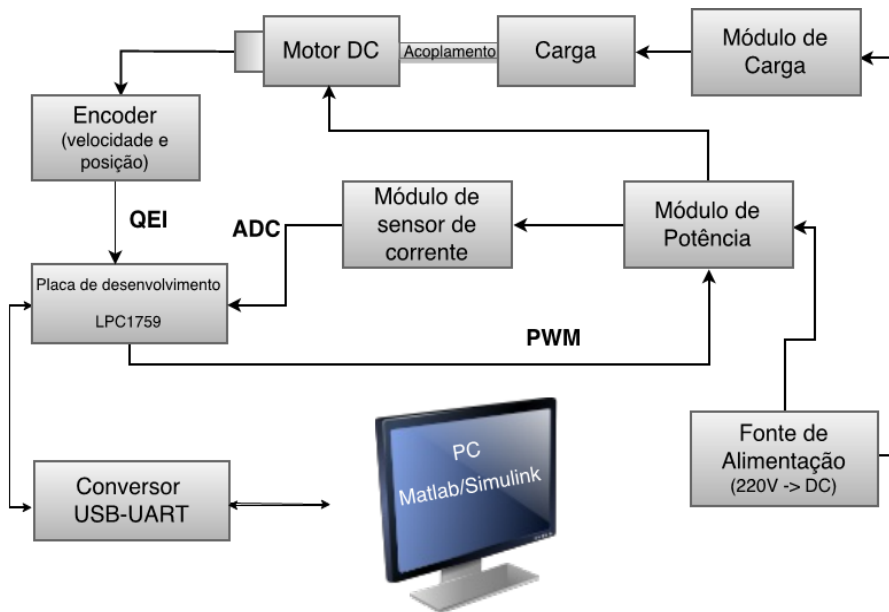


Figura 55 - Esquema geral da plataforma de ensaios

5.1 Estrutura Mecânica

A plataforma de ensaios apresentada na Figura 56 foi construída no âmbito da dissertação referenciada em [7], tendo sido adaptada para o presente trabalho. A plataforma é constituída, mecanicamente, por um motor de ímãs permanentes e um motor de carga. A ligação dos veios dos motores é efetuada através de um acoplamento mecânico. O motor apresentado como motor de carga também pode ser considerado um gerador, sendo que este apresenta um comportamento não linear. A utilização de uma fonte de corrente na saída do motor de carga (gerador) visa testar a qualidade do controlador implementado perante um torque de carga que é aplicado ao motor de controlo.

A utilização de um acoplamento mecânico axialmente flexível na ligação dos dois motores é vantajosa relativamente à utilização de um acoplamento axialmente rígido. O acoplamento mecânico utilizado na plataforma de ensaios permite compensar os desalinhamentos, absorver os picos de torque de arranque do motor DC e também vibrações perigosas. Na Figura 56 mostra-se a representação da plataforma de ensaios com um acoplamento adicional de um codificador de quadratura e um taquímetro. De referir que na presente dissertação não se utiliza o taquímetro, sendo a leitura da velocidade e posição feita pelo codificador.

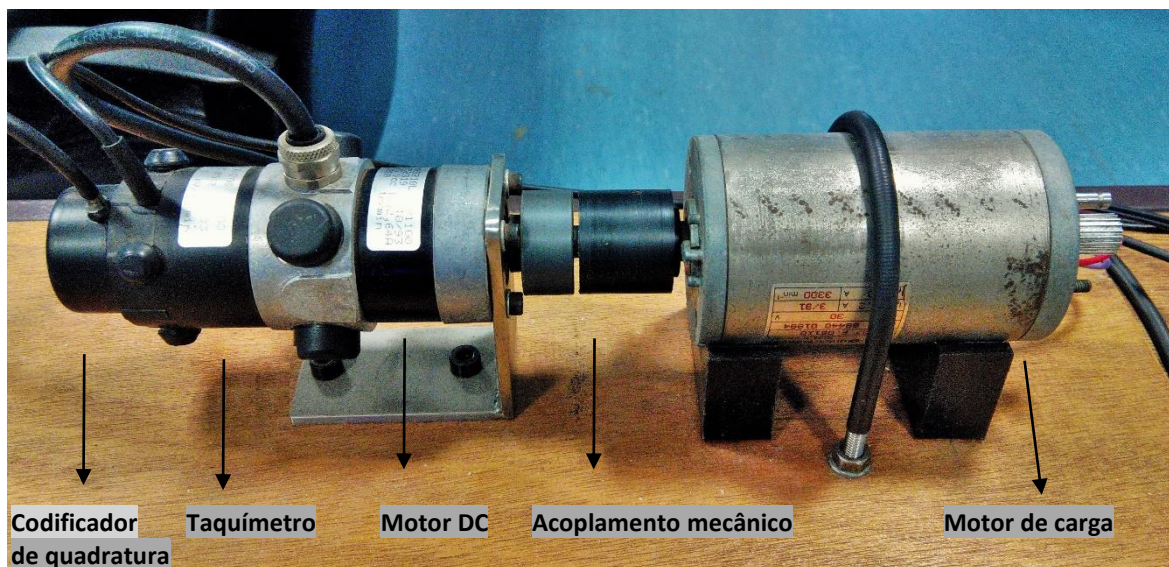


Figura 56 - Plataforma de Ensaio

5.2 Placa de Controlo

Neste tópico apresenta-se a placa de controlo desenvolvida para esta dissertação, sendo constituída pelo módulo de desenvolvimento LPC1759, pela interface de leitura do codificador de quadratura e pela interface de comunicação série. Na Figura 57 mostra-se a placa referenciada neste tópico, tendo sido implementada numa placa de prototipagem do tipo *breadboard*.

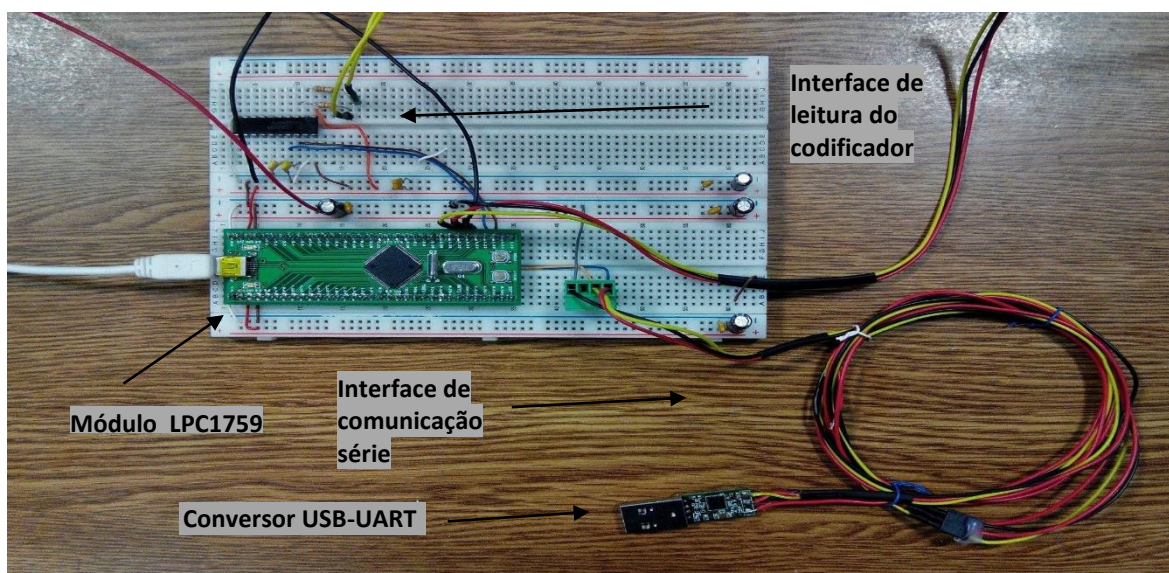


Figura 57 - Placa de controlo do sistema real

Em anexo, apresentam-se os desenhos elétricos referentes à interface de leitura do codificador e comunicação série [Anexo C.5]. É também apresentado o desenho elétrico do módulo de desenvolvimento LPC1759 [Anexo C.1] utilizado na placa *breadboard* que forma a placa de controlo (Figura 57), bem como os seus *layouts* [Anexos C.2, C.3 e C.4]. Para desenho

do circuito elétrico da interface de leitura do codificador de quadratura e da interface de comunicação série, utilizou-se a ferramenta *Orcad Capture* [29].

Para a implementação do leitor dos sinais provenientes do codificador, utilizou-se o circuito integrado (CI) 74AC245PC [30], sendo este um *transceiver* bidirecional com *buffers* de sinal com saída de três estados. Um dos motivos pelo qual se utilizou este *transceiver* foi para proteção da linha de dados contra ruídos externos que possam eventualmente influenciar os níveis lógicos dos sinais. Utilizou-se um módulo interno do microcontrolador para leitura do codificador (QEI), não havendo necessidade de selecionar um circuito integrado.

A interface de comunicação série é utilizada para visualização dos dados obtidos nos ensaios efetuados ao sistema via MATLAB. Utilizou-se um módulo conversor USB (*Universal Serial Bus*) para UART (*Universal Synchronous Receiver/Transmitter*) baseado no circuito integrado FT232RL [31]. Este módulo, representado na Figura 57, serve de interface entre a unidade de controlo e a interface desenvolvida em MATLAB.

Relativamente à seleção da placa de desenvolvimento, optou-se por uma com as características necessárias ao desenvolvimento do controlador do sistema nomeadamente a leitura de sensores, conversão de sinais analógicos em sinais digitais, controlo de motores, fácil integração em sistemas de tempo real, entre outros. Sendo assim, selecionou-se uma placa de desenvolvimento com os seguintes periféricos: um conversor analógico/digital (ADC - *Analog Digital Converter*), uma interface com *encoders* de quadratura (QEI), uma porta série UART e um módulo de PWM dedicado ao controlo de motores (MCPWM). Selecionou-se a placa de desenvolvimento LPC1759 já existente e desenvolvida no Departamento de Eletrónica, Telecomunicações e Informática da Universidade de Aveiro.

A placa selecionada apresenta um microcontrolador da série LPC1700 da empresa NXP Semiconductors [3] baseado na arquitetura ARM Cortex-M3. Para além das características referidas anteriormente, apresentam-se outras:

- . Frequência de relógio do processador até 120MHz;
- . Memória *flash* de 512KB e memória RAM de 64KB;
- . Interface Ethernet, USB *Host/Device*, UART, CAN, SPI, I2C, ADC 12 bits, QEI, MCPWM, Timers, portos de GPIO rápidos;
- . Programação do microcontrolador via porta série RS232 a partir de um *bootloader* integrado;
- . Unidade de proteção de memória (MPU);
- . Baixa latência nas interrupções.

A placa de desenvolvimento selecionada permite o acesso direto aos pinos mais importantes do microcontrolador devido ao seu formato DIP (*Dual-Inline Package*). A conceção de uma placa no formato DIP é vantajosa na medida em que permite uma maior flexibilidade no desenvolvimento dos projetos como por exemplo a utilização de módulos externos ao microcontrolador e a simples montagem em placas *breadboard*.

5.3 Eletrônica de atuação e de medida

Para o acionamento do motor DC pertencente à plataforma de ensaios, utilizou-se um módulo de potência, no âmbito da dissertação referenciada em [7]. Posteriormente, são apresentados os módulos de medição, que foram desenvolvidos nesta dissertação.

5.3.1 Módulo de Potência

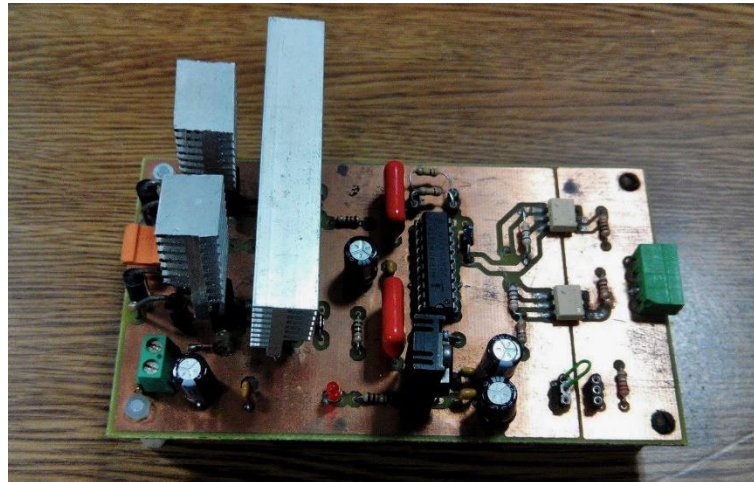


Figura 58 - Módulo de Potência

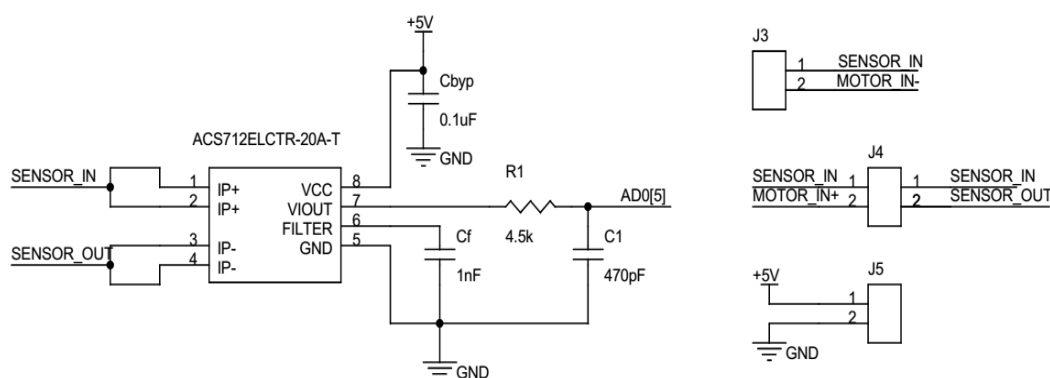
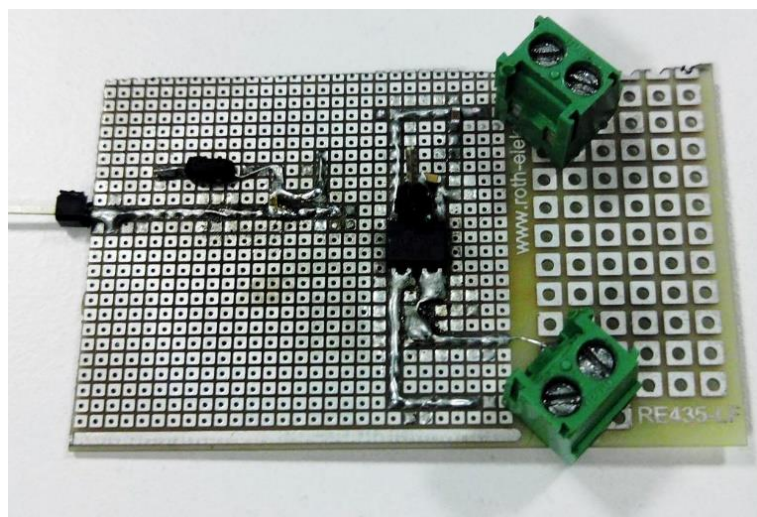
O módulo de potência apresentado na Figura 58 serve de interface entre o microcontrolador e o motor DC de forma a que os sinais PWM gerados pelo microcontrolador permitam o acionamento do motor DC. Este módulo é constituído por um circuito elétrico baseado essencialmente numa ponte H e num *driver* que permite o controlo de cada transistor de potência que faz parte da ponte H [Anexo D]. O circuito de comutação, representado numa topologia em H é constituído por quatro MOSFET's (*Metal Oxide Semiconductor Field Effect Transistor*), sendo estes de canal N.

No módulo de potência, o *driver* responsável pela comutação dos MOSFET's da ponte H é o circuito integrado HIP4081 [32]. O *driver* necessita de uma tensão de alimentação nominal de 12V para o seu funcionamento correto.

Como já referido anteriormente, os sinais PWM são enviados pelo microcontrolador para as entradas do controlador de MOSFET's permitindo o acionamento do motor DC. Esses sinais são sinais digitais compatíveis com a lógica TTL (*Transistor-Transistor Logic*), sendo esta lógica suportada pelo microcontrolador.

5.3.2 Módulo Sensor de corrente

Para a medição da corrente elétrica do sistema motor-carga apresentado no presente capítulo, desenvolveu-se um módulo sensor de corrente composto pelo circuito integrado ACS712 da empresa *Allegro MicroSystems* [33] e pelo circuito de condicionamento de sinal à saída do sensor. De seguida, apresenta-se o sensor de corrente desenvolvido numa placa de circuito impresso perfurada DIP/SMD e, posteriormente o esquema elétrico implementado em *OrCAD* [29].



Para a leitura da corrente foi selecionado o sensor de corrente ACS712ELCTR-20A-T devido ao seu preço atrativo e de tamanho reduzido em encapsulamento SOIC-8. Trata-se de um sensor de efeito de *Hall* que permite a medição de corrente contínua e alternada até $\pm 20A$. Com a decisão do uso do valor da tensão de alimentação do sistema de 12V, a corrente elétrica varia entre $\pm 6A$, aproximadamente, pelo que o sensor de corrente seleccionado adequa-se a esta utilização. Este circuito integrado apresenta outras características tais como uma boa linearidade, imunidade a ruídos eletromagnéticos e um tempo de resposta de 5 μs . No que se refere ao princípio de funcionamento, quando o valor de corrente a medir é nulo (0 A), a saída do sensor (VIOUT) apresenta um *offset* de 2.5V. O sensor de corrente necessita de uma tensão de alimentação de 5 V e a sua saída VIOUT apresenta uma sensibilidade de 100mV/A.

O CI (circuito integrado) ACS712 representado na Figura 60 apresenta um isolamento elétrico entre os pinos de potência (1,2,3 e 4) e os pinos de saída (5 a 8), permitindo assim que o sensor possa ser usado em aplicações onde este isolamento é necessário. O componente ACS712, em configuração de sensor de corrente, deve ser ligado em série com a carga. Neste caso, o sensor de corrente é ligado em série com a entrada MOTOR_IN+ (Figura 60), que corresponde à saída 1 do módulo de potência, representada pelo conector J3.

Para o condicionamento do sinal de saída VIOU, dimensionou-se um filtro passa-baixo do tipo RC com o objetivo de minimizar os efeitos dos ruídos. O dimensionamento do filtro teve em conta a seguinte frequência de corte:

$$f_{corte} = \frac{1}{2\pi R_1 C_1} = 25Hz \quad (5.1)$$

O filtro passa-baixo é colocado imediatamente antes da entrada do canal de ADC (AD0[10]), tendo este uma tolerância máxima de 3.3V. A saída do sensor de corrente representa uma saída analógica de tensão proporcional à corrente elétrica, de acordo com a curva caraterística dada pelo fabricante do sensor ACS712 [33]. Efetuou-se um teste de validação do módulo implementado, utilizando o seguinte esquema elétrico.

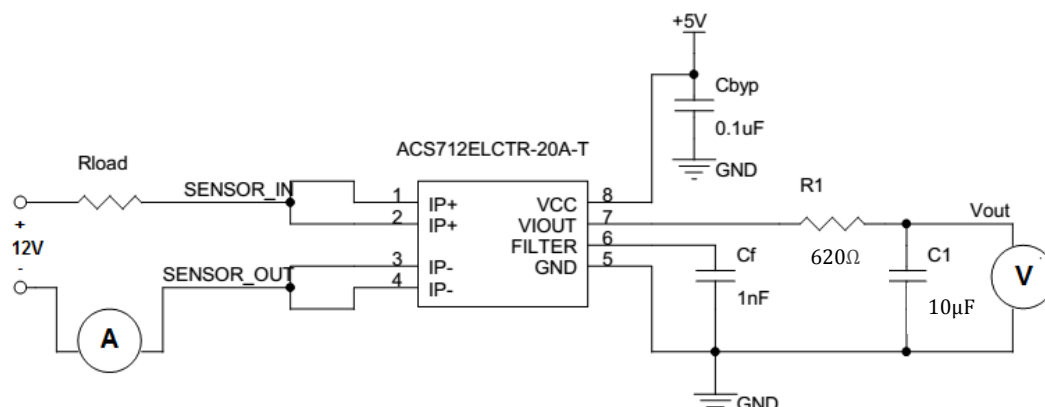


Figura 61 - Esquema elétrico utilizado na validação do módulo de sensor de corrente

Variando a resistência de carga (R_{load}) de 2Ω a 12Ω , mediu-se a corrente elétrica utilizando um amperímetro em série com o pino SENSOR_OUT do módulo e a tensão (V_{out}) com um voltímetro. De seguida, apresentam-se os resultados tabelados do teste efetuado ao módulo referido anteriormente.

Resistência de carga (Ω)	2	2.4	3	4	6	12	2	2.4	3	4	6	12
Tensão (V)	1.97	2.08	2.19	2.28	2.38	2.47	2.67	2.76	2.86	2.96	3.06	3.16
Corrente (A)	-6	-5	-4	-3	-2	-1	1	2	3	4	5	6

Tabela 3 - Teste de validação do módulo de sensor de corrente

De acordo com a tabela anterior, verifica-se que a sensibilidade da saída do módulo é aproximadamente 100mV/A. O módulo de sensor de corrente foi testado considerando a sua caraterística bipolar de medição de corrente. O teste de validação efetuado permite concluir que a variação de corrente elétrica em função da saída analógica (tensão) do módulo apresenta o mesmo comportamento da curva caraterística disponibilizada pelo fabricante do sensor ACS712.

5.3.3 Fonte de Corrente

De modo a testar o comportamento do sistema real bem como validar a concretização dos objetivos para esta dissertação, projetou-se uma fonte de corrente. Pretende-se impor uma variação de torque ao sistema de controlo utilizando uma fonte de corrente ligada à saída do motor de carga. Existem diversas configurações de fonte de corrente constante, sendo as configurações mais básicas: fonte de corrente constante com amplificador operacional (OPAMP), fonte de corrente constante com transístor e fonte de corrente com regulador de tensão. Na presente dissertação, projetou-se uma fonte de corrente de valor constante utilizando um regulador de tensão. Na figura seguinte, apresenta-se o esquema elétrico da fonte de corrente.

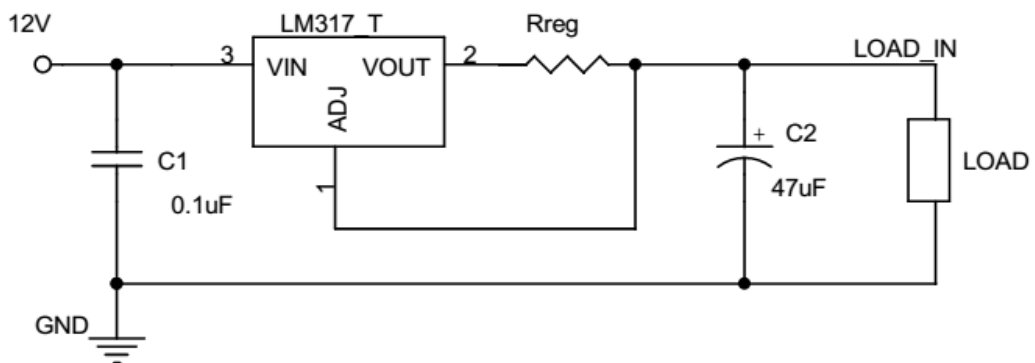


Figura 62 - Esquema elétrico da fonte de corrente com regulador LM317

O circuito integrado utilizado na implementação da fonte de corrente é um regulador linear de tensão LM317_T, da empresa *ON SemiConductor* [34]. A configuração apresentada na Figura 62 permite utilizar o regular de tensão como fonte de corrente constante.

O circuito integrado LM317_T é um regulador de tensão de encapsulamento T0-220 capaz de fornecer uma corrente de saída superior a 1.5A para uma gama de tensão entre 1.2V e 37V, contém proteção interna contra curto-circuito e utiliza componentes adicionais de baixo custo. Na configuração de fonte de corrente, em condições normais de operação, a tensão entre o pino de saída VOUT e o pino ADJ é de valor constante $V_{ref}=1.25V$. Uma vez que a tensão V_{ref} é constante, a corrente de saída fornecida pelo circuito é dependente do valor da resistência de ajuste (R_{reg}). De acordo com os dados do fabricante [34], o valor da resistência de ajuste deve estar compreendido entre $0.8\ \Omega$ e $120\ \Omega$, o que significa que a corrente mínima fornecida pelo circuito é de 10 mA. A saída da fonte de corrente é ligada diretamente ao motor de carga (LOAD), estando esta representada na figura por LOAD_IN. A corrente de saída da fonte apresenta uma relação de proporcionalidade inversa com a resistência de ajuste e é dada pela expressão:

$$I_{out}(A) = \frac{V_{ref}}{R_{reg}} = \frac{1.25}{R_{reg}} \quad (5.2)$$

5.4 Sistema operativo de tempo real

Anteriormente foram especificados todos os componentes de *hardware* que constituem a plataforma de ensaios. A nível de *hardware*, o microcontrolador a ser utilizado pertence à arquitetura ARM Cortex-M3. Sendo um dos objetivos desta dissertação o desenvolvimento de uma aplicação de controlo do sistema sob um *kernel* de tempo real, foi necessário escolher um RTOS (*Real-Time Operating System*) que suportasse a arquitetura referida. Um sistema de tempo real é concebido para satisfazer a necessidade de uma aplicação de tempo real garantindo o cumprimento dos requisitos temporais (*deadlines*). Pode ser classificado como *soft real-time* ou *hard real-time*. Num sistema *soft real-time* o incumprimento não frequente dos *deadlines* não apresentam consequências graves mas degrada a qualidade do sistema. Para um sistema *hard real-time*, o incumprimento dos requisitos temporais tem consequências graves na qualidade do sistema deixando de existir determinismo.

No mercado atual, existem diversos sistemas operativos *open source* de tempo real vocacionados para sistemas embebidos que se encontram em desenvolvimento permanente [35]: FreeRTOS, ChibiOS, CoActionOS, entre outros. Na presente dissertação optou-se pelo sistema operativo FreeRTOS devido à sua elevada portabilidade e pela sua simplicidade [4], podendo ser utilizado por diversas arquiteturas de processadores (AVR32, ARM Cortex-M3, STM32, PIC32, 8051, entre outras). É um sistema capaz de proporcionar um ambiente de execução multitarefa e *multithread* (capacidade de executar várias tarefas em simultâneo). O FreeRTOS também é caracterizado por disponibilizar um conjunto de bibliotecas que permitem um escalonamento preemptivo [4], podendo deste modo parar a execução de uma tarefa para executar outra com prioridade superior. Existe a possibilidade de duas ou mais tarefas terem a mesma prioridade de execução. Também possui mecanismos de comunicação e sincronização entre tarefas, como por exemplo semáforos, filas de mensagens e *mutexes*. Estes mecanismos fazem parte de uma Interface facultada pelo FreeRTOS, denominada de Interface de Programação de Aplicativos (API).

De salientar que a quantidade de memória utilizada por este sistema operativo é bastante reduzida (4kb a 9Kb) e esta é configurável pelo utilizador permitindo assim uma rentabilização da memória. O código fonte deste RTOS encontra-se implementado em C e, como já referido, tem uma licença livre. Juntamente com o código, é possível obter de forma gratuita documentação com a especificação do sistema operativo e também uma *demo* com aplicações exemplo, permitindo assim ao utilizador uma melhor compreensão podendo desenvolver aplicações de forma eficaz.

5.5 Software/ Firmware

Neste tópico descreve-se o *software* implementado para a identificação paramétrica e controlo do sistema real, acompanhado de fluxogramas. A implementação de código no microcontrolador LPC1759 foi feita recorrendo à ferramenta SublimeText em ambiente Linux. Trata-se de um editor de texto e código multi-plataforma caracterizado pela sua simplicidade e flexibilidade, permitindo facilitar o desenvolvimento de *software/firmware* em sistemas embebidos. Esta ferramenta oferece recursos que permitem aumentar a produtividade de quem desenvolve, como por exemplo, a função auto-completar, *snippets*, *minimap*, entre outros. Utilizou-se um script *makefile* com sub-rotinas criadas de modo a executar as tarefas de *build* e *program* no microcontrolador.

Primeiramente procedeu-se à instalação de um programa na memória *flash* do microcontrolador LPC1759 denominado *bootloader* USB. A função de um *bootloader* é controlar a comunicação com um PC permitindo reprogramar o microcontrolador sem ter de recorrer a um programador físico. Posteriormente, procedeu-se à implementação dos *device drivers* necessários para a implementação do software de identificação e controlo.

Relativamente ao desenvolvimento de *software/firmware*, a empresa ARM disponibiliza uma Interface de Programação de Aplicativos (API) específica para a arquitetura de processadores ARM Cortex-M, denominada de CMSIS (*Cortex Microcontroller Software Interface Standard*). Esta camada de abstração de *hardware* disponibiliza uma interface de software simples do processador aos periféricos, sistemas operativos de tempo real (RTOS) e ao *middleware*. A utilização do CMSIS é vantajosa uma vez que permite a reutilização de *software*, reduzindo a curva de aprendizagem aos novos programadores [36]. Na presente dissertação, para a implementação do *firmware* utilizou-se a o CMSIS para o microcontrolador da série LPC1700 com as funções que permitem aceder aos registos do *core* e aos seus periféricos.

5.5.1 Device Drivers (*Firmware*)

A implementação dos *device drivers* essenciais ao funcionamento do sistema é efetuada numa camada de baixo nível, sendo responsáveis pela comunicação com os periféricos do microcontrolador de modo a permitir o seu funcionamento. De salientar que a configuração dos *device drivers* é efetuada de acordo com o CMSIS, sendo independente da API disponibilizada pelo FreeRTOS. Apresentam-se de seguida as configurações efetuadas dos periféricos do microcontrolador LPC1759 [3]:

- **UART (*Universal Asynchronous receiver/transmitter*):** permite a comunicação entre a interface GUI desenvolvida em MATLAB e o microcontrolador de uma forma dinâmica. Através deste módulo, são enviados para o PC os dados da posição, velocidade, corrente elétrica e sinal de controlo u obtidos nos ensaios. Este módulo foi configurado utilizando o canal 0 da UART com um *baudrate* de 115200.
- **MCPWM (*Motor Control Pulse Width Modulation*):** o módulo referido permite o acionamento do motor DC. A frequência de relógio selecionada para este periférico foi de 120MHZ, tendo sido utilizadas as saídas A e B do canal 1 (MCOA1 e MCOB1) referentes ao microcontrolador. Para a configuração do período e da largura do pulso utilizaram-se registos MCPER e MCPW, respetivamente. Definiu-se uma frequência de PWM de valor 250 HZ uma vez que utilizando elevadas frequências gera perdas na comutação dos MOSFET's da ponte H. De acordo com a frequência de PWM definida anteriormente, os registos MCPER e MCPW são dados pelas expressões:

$$MCPER = \frac{f_{relógio}}{f_{pwm}} = \frac{120MHz}{250Hz} = 480k \text{ (ticks de relógio)} \quad (5.5)$$

$$MCPW = \frac{MCPER * duty_cycle}{100} \text{ (ticks de relógio)} \quad (5.6)$$

O período do pulso é de valor constante enquanto que a largura de pulso varia de acordo com o valor percentual de *duty cycle*. O *duty cycle* representa a percentagem de tempo que o circuito que alimenta o motor fica a *on* (ligado). Este parâmetro é obtido diretamente do sinal de controlo cujo valor depende dos ganhos de realimentação de estados e dos valores da posição e velocidade lidos pelo codificador de quadratura e da corrente elétrica lida pelo sensor de corrente. De salientar que tensão aplicada ao motor é diretamente proporcional ao *duty cycle*.

- **QEI (*Quadrature Encoder Interface*):** trata-se de um periférico interno do microcontrolador LPC1759 sendo utilizado para a leitura dos sinais provenientes do codificador. O *device driver* que permite a comunicação entre o codificador e o microcontrolador foi configurado para funcionar a uma frequência de 30MHz. Este periférico permite a captura de posição e de velocidade de forma independente. Com a configuração do QEI em modo de fase de quadratura (registo SIGMODE = 1), o pino PHA funciona como sinal de direção e o pino PHB como sinal de relógio para os contadores. A captura de posição é feita através de um registo que funciona como um contador de posição (registo QEIPPOS), tendo sido configurado um outro registo para definir a posição máxima (QEIMAXPOS) do contador. Por outro lado, a captura de velocidade é feita por um timer programável (registo QEITIME) e um registo de captura (QEICAP). Quando o timer entra em *overflow*, o valor do registo do contador de velocidade (QEIVEL) é armazenado no registo de captura. Com isto, o registo QEIVEL é colocado a zero e é reiniciado o timer com o valor do registo QEILOAD. Configurou-se este registo de acordo com a seguinte expressão:

$$QEILOAD = \frac{f_{relogio}}{f_{qei}} = \frac{30MHz}{50Hz} = 600k \text{ (ticks de relógio)} \quad (5.7)$$

- **ADC (*Analog Digital Converter*):** este conversor permite a conversão de um sinal analógico para um sinal digital. Neste caso, utilizou-se o periférico interno do microcontrolador para efetuar a medição de corrente. Para isso, é feita a leitura do sinal de entrada analógico (saída VIOUT do módulo sensor de corrente) e de seguida é a conversão para um sinal digital por parte do periférico interno do microcontrolador. A ADC utilizada apresenta uma resolução de 12 bits. O canal da ADC utilizado para efetuar a amostragem do sinal analógico foi o canal 5 e este processo foi efetuado à frequência de 150kHz, sendo esta frequência permitida pelo microcontrolador LPC1759. Como já referido anteriormente, a frequência de corte do filtro passa-baixo do módulo de sensor de corrente (expressão 5.1) é aproximadamente 25Hz. Dimensionou-se o filtro passa-baixo de modo a evitar o aparecimento de *aliasing* no processo de amostragem. Por outro lado, tem-se a frequência de *Nyquist* que corresponde a metade da frequência de amostragem. Uma vez que a frequência de corte do filtro é aproximadamente igual à frequência de *Nyquist* ($f_{Nyquist} = 25Hz$) são atenuadas as frequências superiores a 25Hz. No entanto, efetuaram-se testes preliminares de leitura da ADC, verificando-se o aparecimento de *glitches* durante o processo de amostragem. Após alguns testes e feita uma pesquisa no fórum do fabricante da arquitetura ARM [37], concluiu-se que os conversores analógicos digitais da série LPC17XX apresentam uma baixa eficiência e um comportamento instável na leitura com o aparecimento de *aliasing*. A solução para este problema foi efetuar uma

filtragem por *software*. Este filtro consistiu em efetuar 100 leituras de ADC seguido de uma ordenação destes valores num *array*. Por fim, efetuou-se uma média dos 10 valores centrados do *array*.

5.5.2 Tarefas

Anteriormente apresentaram-se as configurações dos *device drivers* que permitem o funcionamento do microcontrolador e a comunicação com os módulos apresentados neste capítulo. No sistema real apresentado, implementaram-se duas tarefas: uma para identificação dos parâmetros e outra para implementação do controlador do sistema utilizando realimentação das variáveis de estado. Utilizou-se a função pertencente à API do FreeRTOS para a criação de uma nova tarefa, denominada de *xTaskCreate*. Trata-se de uma função que apresenta como argumentos de entrada o nome da tarefa, a prioridade da tarefa, o tamanho da *stack* de dados, entre outros [4]. Para iniciar o escalonador de tarefas é utilizada a função *vTaskStartScheduler*.

Os fluxogramas que descrevem as tarefas de identificação e de controlo do sistema real encontram-se nos Anexos B.1 e B.2, respetivamente.

A tarefa *vSysId* permite fazer a identificação paramétrica do sistema com uma frequência de execução de 50Hz. Primeiramente, na rotina principal é feita a configuração dos *device drivers* seguido da execução da tarefa *vSysId*, sendo esta iniciada após receção via porta série do comando de início (*sysid_com*) por parte da interface GUI. Uma vez que a tarefa é periódica, utilizou-se a funcionalidade do FreeRTOS que permite a execução de uma tarefa a uma frequência fixa (*vTaskDelayUntil*). É executado um ciclo de amostragem à frequência de 50Hz com um valor máximo de amostras *N*. Em cada ciclo é feita a leitura do QEI (codificador), é executado o algoritmo de identificação *RLS* tendo em conta a leitura da saída do sistema e também é feita a atuação do sistema por MCPWM utilizando o sinal de entrada (sinal de excitação). No final de cada ciclo são enviados os dados via porta série para a interface GUI. Para a identificação do modelo da velocidade do sistema efetuou-se a leitura da velocidade através do codificador (sinal de saída). Por outro lado, para identificar o modelo da posição, utilizou-se como sinal de saída a posição. No capítulo 6 é feita uma descrição mais detalhada do processo de identificação do sistema sendo acompanhada dos resultados obtidos dos ensaios.

A tarefa implementada para controlar o sistema usando realimentação de estados é denominada de *vSysControl*. Primeiramente, na rotina principal configuraram-se os *device drivers* e criou-se a tarefa *vSysControl* utilizando a funcionalidade *xTaskCreate* do FreeRTOS, sendo executada após a inicialização do escalonador de tarefas. Assim que a tarefa é iniciada, é necessário proceder-se à configuração dos requisitos para a resposta transitória do sistema utilizando a interface GUI desenvolvida em MATLAB. Após a configuração, é feito o cálculo dos ganhos de realimentação, sendo o processamento feito do lado do microcontrolador. A execução do ciclo de controlo é feita após envio do comando de início (*syscontrol_com*) via porta série da interface GUI para o microcontrolador. A tarefa de controlo é executada com uma periodicidade de 50Hz através de um ciclo de amostragem, sendo definido um valor máximo de *N* amostras. O ciclo de controlo (amostragem) do sistema é iniciado com a leitura do sinal de referência e de seguida são efetuadas as leituras dos sensores e calculado o sinal de controlo. O passo seguinte consiste em executar a função de atuação do sistema por MCPWM e, por fim, são enviados os dados via porta série para a interface GUI em MATLAB. O

código desenvolvido dentro do ciclo de controlo varia de acordo com o tipo de controlo que se pretende: controlo de velocidade ou posição.

Na implementação do controlador de velocidade são executadas as funções de leitura da velocidade e de corrente enquanto que no caso do controlador de posição utilizam-se as funções de leitura da velocidade e da posição. Para além destas diferenças na implementação entre o ciclo de controlo da velocidade e posição, o cálculo do sinal de controlo também é diferente, estando dependente das variáveis de estado que são utilizadas e dos ganhos de realimentação.

A velocidade obtida pelo codificador de quadratura [3], expressa em rpm, depende de parâmetros como a frequência de relógio do periférico, o registo do contador de velocidade (QEICAP), do número de pulsos por revolução do codificador (PPR) e do número de transições nos canais PHA e PHB do codificador.

$$velocidade_{encoder} = \frac{f_{relogio_{qei}} * QEICAP * 60}{QEILOAD * PPR * edges} = \frac{30M * QEICAP * 60}{600K * 500 * 2} \quad (5.8)$$

A leitura da posição através do *codificador* é feita com recurso à seguinte expressão:

$$posição_{encoder} = \frac{QEIPOS * 360}{PPR} = \frac{QEIPOS * 360}{500} = QEIPOS * 0.72 \quad (5.9)$$

Por fim, a expressão que permite obter a corrente lida pelo sensor ACS712 descrito no tópico 5.3.2 é:

$$corrente_{sensor} = \frac{tensão_{sensor} - tensão_{offset}}{sensibilidade_{sensor}} \quad (5.10)$$

Na expressão anterior, a tensão do sensor resulta da conversão do valor de ADC, a tensão *offset* utilizada é de valor 2.5V e a sensibilidade do sensor é de 100mV/A, como já referido anteriormente na descrição do módulo sensor de corrente. A tensão do sensor é apresentada pela seguinte expressão:

$$tensão_{sensor} = \frac{valor_{ADC}}{resolução_{ADC}} * tensão_{ref} \quad (5.10)$$

A expressão 5.10 permite converter o valor de ADC para o valor de tensão, em que a resolução da ADC é de valor 4096 e a tensão de referência do conversor analógico digital é de 3.3V.

Capítulo 6 Identificação do Modelo Dinâmico

6.1 Considerações gerais sobre Identificação de Sistemas

Em controlo, a identificação consiste, basicamente, na obtenção de um modelo que representa a informação contida nos sinais de entrada e de saída de um determinado sistema [8]. Estes dados devem ser recolhidos durante a realização de procedimentos previamente determinados, onde é possível definir os sinais de entrada e saída, os intervalos de amostragem bem como seleccionar o tipo de sinal aplicado à entrada de modo a que a informação nele contida seja máxima. O modelo do sistema é obtido de acordo com os seguintes requisitos fundamentais: os dados experimentais, uma estrutura de modelos e um critério de seleção do melhor modelo.

Os dados experimentais utilizados para identificar modelos de sistemas dinâmicos podem ser obtidos a partir de medições da saída y e da entrada u . É necessário que o sinal de entrada u possua características que lhe permitam estimular suficientemente todos os modos de funcionamento do sistema e posteriormente recolher um conjunto de dados com informação suficiente de modo a atingir os objetivos pretendidos.

Para além dos dados experimentais, é necessário escolher uma estrutura de modelos. Existem duas classes de estruturas de modelos, uma descrita sob a forma polinomial e função de transferência e a outra é descrita sob a forma de espaço de estados. As estruturas do modelo referidas são apresentadas com mais detalhe numa fase posterior do documento.

Por fim, a seleção do melhor modelo do sistema é feita de acordo com um método de identificação, que consiste na minimização de um critério de erro [8]. Este critério de erro está relacionado com a precisão com que o modelo selecionado consegue reproduzir os dados e com isto, apresentar um comportamento o mais próximo possível do comportamento real do sistema. A figura seguinte apresenta o diagrama de blocos do processo de identificação de sistemas.

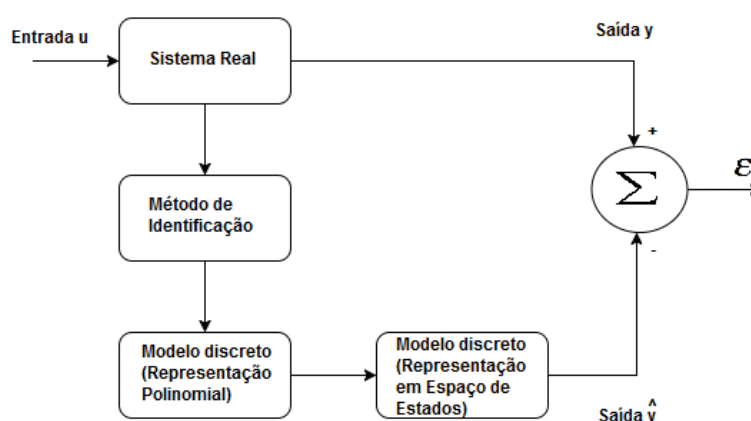


Figura 63 - Processo de Identificação de Sistemas

A figura anterior mostra na forma de diagrama de blocos o processo de identificação de um sistema real. É aplicado à entrada do sistema um sinal de excitação u e através de um método designado de método de identificação obtém-se uma estimativa dos parâmetros do sistema. A utilização de um modelo discreto com representação polinomial e a posterior passagem para um modelo discreto por espaço de estados faz parte de um dos objetivos da presente dissertação. Por comparação da saída real do sistema y e da saída estimada \hat{y} do modelo obtém-se o erro de estimação ε .

Nos tópicos que se seguem, faz-se uma abordagem teórica de todo o processo de identificação de um sistema real e, por fim, apresentam-se os resultados dos ensaios efetuados ao motor DC real apresentado no capítulo 5.

6.2 Sinal de entrada do sistema

O sinal aplicado à entrada do sistema dinâmico é um sinal de excitação que deve cumprir certos requisitos de modo a que o sistema seja bem identificado. Este sinal deve excitar persistentemente a dinâmica do sistema durante o período de identificação. Esta característica do sinal de entrada faz com que seja conhecido como sinal de excitação persistente [38]. Como exemplos de sinais de excitação persistente tem-se o ruído branco, uma sequência binária pseudo aleatória (PRBS - *Pseudo Random Binary Sequence*) e um sinal em forma de degrau. Dos sinais apresentados, o ruído branco é o melhor sinal de excitação uma vez que apresenta um conjunto de frequências muito variadas. No entanto, se houver a restrição por parte do microcontrolador no que respeita à sua capacidade de processamento, torna-se difícil gerar este tipo de sinais. Como alternativa ao sinal de ruído branco, utilizam-se sinais do tipo PRBS.

Comparativamente aos sinais de ruído branco, os sinais PRBS são mais fáceis de gerar e controlar e permitem obter um modelo do sistema que apresente as características dinâmicas que se aproximem das características dinâmicas do sistema real. Este tipo de sinal, que é aplicado na entrada u do sistema apresenta propriedades que podem ser consultadas em [38].

6.3 Modelos discretos do sistema

Conforme referido no tópico 6.1 existem duas classes de estruturas de modelos que representam um dado sistema dinâmico e que são utilizadas aquando do processo de identificação. É dado uma maior ênfase às classes de estruturas de modelos sob a forma de espaço de estados, uma vez que se trata de um dos objetivos da presente dissertação.

No presente tópico, as estruturas de modelos apresentadas referem-se a expressões matemáticas parametrizadas que descrevem o comportamento dinâmico original do sistema real. Os parâmetros destas estruturas matemáticas são ajustados por algoritmos de estimação a partir dos dados de entrada-saída medidos ou por algoritmos de estimação das variáveis internas do sistema (modelos em espaço de estados). Estes algoritmos de estimação são abordados numa fase posterior.

6.3.1 Classes de estruturas de modelos com representação polinomial

Sendo um dos objetivos deste capítulo a obtenção de um modelo linear discreto de um sistema, este é representado pela seguinte equação [8]:

$$y(k) = G(q^{-1})u(k) + H(q^{-1})e(k) \quad (6.1)$$

Na equação anterior a função de transferência discreta do sistema é representada pelo termo $G(q^{-1})$, as perturbações do sistema são representadas por $H(q^{-1})$, o ruído branco é representado por $e(k)$ e, por fim, o operador atraso por q^{-1} .

Existe uma forma genérica de modelos polinomiais, que é definida pela seguinte equação:

$$A(q^{-1})y(k) = \frac{B(q^{-1})}{F(q^{-1})}u(k) + \frac{C(q^{-1})}{D(q^{-1})}e(k) \quad (6.2)$$

Partindo da estrutura genérica de um modelo discreto dado pela equação anterior, pode-se obter diferentes classes de estruturas de modelos, através da parametrização dos polinómios A , B , C , D e F .

A representação mais simples de um modelo na forma polinomial é dada pela classe de estruturas de modelos do tipo ARX (*AutoRegressive with eXogenous input*). O modelo do tipo ARX, como já referido, é usado em modelos lineares e representa-se pela seguinte equação:

$$y(k) = \frac{q^{-t_d}B(q^{-1})}{A(q^{-1})}u(k) + \frac{1}{A(q^{-1})}e(k) \quad (6.3)$$

em que t_d representa o tempo morto no domínio discreto. Os polinómios $A(q^{-1})$ e $B(q^{-1})$ são descritos pelas equações 6.4 e 6.5, respetivamente:

$$A(q^{-1}) = 1 + a_1q^{-1} + \dots + a_{n_a}q^{-n_a} \quad (6.4)$$

$$B(q^{-1}) = b_1q^{-n_d} + b_2q^{-(n_d+1)} + \dots + b_{n_b}q^{-(n_d+n_b-1)} \quad (6.5)$$

em que n_a , n_b e n_d representam a ordem dos polinómios A , B e D , respetivamente. O modelo ARX pode ainda ser expresso pela seguinte equação linear de diferenças [8]:

$$y(k) + a_1y(k-1) + \dots + a_{n_a}y(k-n_a) = b_1u(k-n_d) + \dots + b_{n_b}u(k-(n_d+n_b-1)) + e(k) \quad (6.6)$$

O modelo ARX, apresentado anteriormente, é um modelo vantajoso no sentido em que os seus parâmetros são facilmente obtidos [8] utilizando o método de identificação dos mínimos quadrados, método este que é apresentado posteriormente nesta dissertação. A estrutura do modelo ARX, abordada anteriormente, é um dos exemplos mais simples de estruturas de modelos. Como outros exemplos, tem-se as seguintes estruturas: ARMAX (*AutoRegressive Moving Average with eXogenous signal*), FIR (*Finite Impulse Response*), OE (*Output Error*) e BJ (*Box-Jenkins*). A descrição das estruturas dos modelos referidos podem ser consultadas em [39].

6.3.2 Obtenção do modelo de espaço de estados a partir da representação polinomial do sistema

Uma vez que se pretende implementar um controlador por espaço de estados do motor DC apresentado no capítulo anterior, torna-se necessário obter um modelo discreto do sistema real a controlar. Partindo da equação de diferenças apresentada na expressão 6.6, que representa o modelo ARX, é possível obter uma representação de espaço de estados do sistema discreto. Segundo Ogata [22], um sistema discreto pode ser representado por espaço de estados na seguinte forma:

$$\begin{bmatrix} x_1(k+1) \\ x_2(k+1) \\ \vdots \\ x_{n-1}(k+1) \\ x_n(k+1) \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & \dots & 0 \\ 0 & 1 & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & 1 \\ -a_n & -a_{n-1} & -a_{n-2} & \dots & -a_1 \end{bmatrix} \begin{bmatrix} x_1(k) \\ x_2(k) \\ \vdots \\ x_{n-1}(k) \\ x_n(k) \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix} u(k) \quad (6.7)$$

$$y(k) = [-b_n - a_n b_0 \quad b_{n-1} - a_{n-1} b_0 \quad \dots \quad b_1 - a_1 b_0] \begin{bmatrix} x_1(k) \\ x_2(k) \\ \vdots \\ x_n(k) \end{bmatrix} + b_0 u(k) \quad (6.8)$$

A representação por espaço de estados do sistema apresentada anteriormente é denominada de forma canónica controlável [22]. A expressão 6.7 representa a equação de estado do sistema e a expressão 6.8 representa a equação de saída do sistema. Os parâmetros a_n e b_n representam os parâmetros de identificação, resultantes do método de identificação do sistema. A abordagem aos métodos de identificação de sistemas é feita posteriormente neste documento. A ordem do sistema é dada pela variável n .

A forma canónica controlável representada nas equações 6.7 e 6.8 corresponde à forma genérica de espaço de estados das equações 3.58 e 3.60, respetivamente. De referir que, para além da forma canónica controlável existem outras formas de representação do sistema discreto. São elas a forma canónica observável, forma canónica diagonal e a forma canónica de *Jordan* [22], que não fazem parte do objetivo desta dissertação.

6.4 Métodos de Identificação Paramétrica de Sistemas

Conforme apresentado na Figura 63, uma das etapas no processo de identificação de um sistema consiste na utilização de um método de identificação. O método de identificação também é designado de identificador. O método dos mínimos quadrados, também designado em inglês de LS (*Least-Squares*) é um dos métodos utilizados para aproximar o comportamento do modelo estimado ao comportamento do sistema real. Para mais detalhes, consultar [40].

A expressão seguinte é designada de equação de regressão e permite obter o valor estimado da saída $\hat{y}(k)$.

$$\hat{y}(k) = \theta^T \varphi(k) \quad (6.9)$$

Na equação anterior, o vetor θ contém os parâmetros a estimar do sistema e $\varphi(k)$ representa o vetor dos regressores. Este vetor representa um conjunto de variáveis

conhecidas, sendo estas versões atrasadas dos sinais de entrada e de saída do sistema. Os vetores θ e φ são dados na forma [40]:

$$\theta = [\theta_1 \ \theta_2 \ \dots \ \theta_n] \quad (6.10)$$

$$\varphi(k) = [\varphi_1(k) \ \varphi_2(k) \ \dots \ \varphi_N(k)] \quad (6.11)$$

em que n representa o número de parâmetros do modelo do sistema e N o número de observações recolhidas do sistema. Para mais detalhes, consultar [40]. A implementação recursiva do método dos mínimos quadrados designa-se de *RLS* (*Recursive Least-Squares*). De seguida faz-se uma breve apresentação deste método.

6.4.1 Método dos Mínimos Quadrados recursivo

O método de identificação dos mínimos quadrados caracteriza-se por ser um método simples de se implementar. No entanto, para se efetuar a estimação dos parâmetros do sistema em tempo-real, o método mais adequado é o *RLS*, sendo bastante utilizado em Controlo Adaptativo. Na presente dissertação efetua-se uma identificação prévia do modelo do sistema utilizando o método recursivo. No entanto, implementa-se um controlador por realimentação de estados com base no modelo previamente identificado devido a limitações de memória do microcontrolador LPC1759. A metodologia referida anteriormente serve de “ponte” para trabalho futuro.

De seguida, apresentam-se as equações utilizadas pelo método *RLS* [40] [20].

$$\hat{\theta}(k) = \hat{\theta}(k-1) + K(k)\varepsilon(k) \quad (6.12)$$

$$\varepsilon(k) = y(k) - \varphi^T(k)\hat{\theta}(k-1) \quad (6.13)$$

$$K(k) = \frac{P(k-1)\varphi(k)}{1 + \varphi^T(k)P(k-1)\varphi(k)} \quad (6.14)$$

$$P(k) = (I - K(k)\varphi^T(k)P(k-1)) \quad (6.15)$$

O vetor $\hat{\theta}(k)$ contém parâmetros que são desconhecidos e que correspondem aos parâmetros a e b apresentados nos tópicos 6.3.1 e 6.3.2 relativamente aos modelos discretos que representam um sistema real.

Na equação 6.13, a variável ε representa o erro de estimação, a variável y representa a saída do sistema real e k é o número da iteração utilizada no processo de identificação do sistema. A matriz P designa-se por matriz de covariância em que a dimensão desta é determinada pelo número de parâmetros n do modelo. Na versão recursiva do método dos mínimos quadrados, é necessário aplicar uma condição inicial à matriz de covariância. Na prática a matriz de covariância no instante inicial é do tipo ρI [41]. A utilização de um valor de ρ pequeno implica alguma confiança no estimador e permite que ele varie de forma lenta enquanto que a utilização de um valor elevado de ρ faz com que haja variações rápidas na estimativa [41].

Comparativamente ao método não-recursivo dos mínimos quadrados (LS), o método recursivo *RLS* não necessita de guardar todos os valores regressores permitindo libertar espaço em memória e economizar em termos de tempo de processamento.

O método recursivo dos mínimos quadrados baseia-se na estimativa do valor dos parâmetros do sistema tendo em conta o valor anterior, isto é, permite a estimação dos parâmetros no instante k com base na estimação efetuada no instante $k - 1$. Quando se tem um sistema cujos parâmetros variam ao longo do tempo, utilizam-se métodos baseados no método *RLS* com a introdução de um fator, designado de fator de esquecimento. São eles os seguintes: método dos mínimos quadrados com esquecimento direcional, método dos mínimos quadrados pesados com fator de esquecimento, método de seleção do fator de esquecimento em função do erro de estimativa e método dos mínimos quadrados com reinicialização da matriz de covariância [41]. De seguida descreve-se o método de identificação do sistema apresentado no capítulo 5 a ser utilizado na presente dissertação, mais concretamente o método dos mínimos quadrados com esquecimento direcional.

6.4.1.1 Mínimos Quadrados com esquecimento direcional

Este método baseia-se no método recursivo dos mínimos quadrados (*RLS*) utilizando um fator de esquecimento, representado por λ . Este parâmetro indica a rapidez com que a informação mais antiga é esquecida e pode apresentar valores que estejam dentro do intervalo $[0,1]$. O método apresentado neste tópico permite corrigir diversos problemas apresentados por outros métodos dos mínimos quadrados com fator de esquecimento. Por exemplo, a utilização do método dos mínimos quadrados pesados com fator de esquecimento apresenta limitações resultantes do aparecimento de uma particularidade, designada de *estimator windup* [41]. O problema do *estimator windup* verifica-se com o aumento dos valores da matriz de covariância quando não existe variação de parâmetros, podendo resultar no aumento do erro de estimação uma vez que o estimador vai perdendo os valores pertencentes ao vetor dos parâmetros estimados. Sendo assim, no método apresentado neste tópico o fator de esquecimento é aplicado de modo a manter constante uma quantidade de informação sempre que os parâmetros não variem.

A implementação deste método baseia-se nas seguintes equações [41] que apresentam semelhanças com as equações apresentadas anteriormente para o método *RLS*, sem fator de esquecimento. São elas:

$$\mathbf{K}(k) = \frac{\mathbf{P}(k-1)\boldsymbol{\varphi}(k)}{1 + \boldsymbol{\varphi}^T(k)\mathbf{P}(k-1)\boldsymbol{\varphi}(k)\alpha(k)} \quad (6.16)$$

$$\mathbf{P}(k) = (\mathbf{I} - \alpha(k)\mathbf{K}(k)\boldsymbol{\varphi}^T(k)\mathbf{P}(k-1)) \quad (6.17)$$

$$\alpha(k) = \begin{cases} \lambda - \frac{1 - \lambda}{\boldsymbol{\varphi}(k)\mathbf{P}(k-1)\boldsymbol{\varphi}^T(k)} & \text{se } \boldsymbol{\varphi}(k)\mathbf{P}(k-1)\boldsymbol{\varphi}^T(k) > 0 \\ 1 & \text{se } \boldsymbol{\varphi}(k)\mathbf{P}(k-1)\boldsymbol{\varphi}^T(k) = 0 \end{cases} \quad (6.18)$$

As equações 6.12 e 6.13, que representam o vetor dos parâmetros estimados e o erro de estimação, respetivamente, apresentam-se da mesma forma no método de mínimos

quadrados com fator de esquecimento direcional. Por fim, este método apresenta um algoritmo de complexidade superior ao do algoritmo *RLS*, sem fator de esquecimento.

6.5 Resultados Experimentais

Neste tópico apresentam-se os resultados relacionados com a identificação do sistema que se pretende controlar e, posteriormente são apresentados também os modelos discretos da velocidade e posição na forma polinomial e na forma de espaço de estados. O sistema a controlar é o conjunto motor-carga apresentado no capítulo 5.

Na identificação dos parâmetros do sistema real foi utilizado o método recursivo dos mínimos quadrados (*RLS*) com esquecimento direcional, apresentado no tópico 6.4.4.1. Todo o processo de identificação do sistema foi feito recorrendo ao microcontrolador LPC1768 utilizando a linguagem de programação C sob o sistema operativo de tempo real FreeRTOS.

Para visualização em tempo real dos dados dos ensaios efetuados ao sistema, implementou-se uma interface gráfica de interação com o utilizador (GUI) em MATLAB [Anexo A.1]. Esta interface permite receber os dados que são enviados por linha série (UART) do microcontrolador para o PC, onde foi desenvolvida a GUI.

6.5.1 Identificação do modelo discreto da velocidade

Neste tópico apresentam-se os resultados dos ensaios em malha aberta do motor DC, sendo que na entrada do motor DC aplicou-se um sinal do tipo PRBS. Este sinal de excitação baseia-se em valores aleatórios compreendidos entre 50 e 100% com atualização a cada 2 segundos, sendo que estes valores são proporcionais aos valores de tensão aplicados ao motor. Implementou-se o método de identificação *RLS* de segunda ordem com um fator de esquecimento direcional de 0.98. De seguida, apresentam-se os resultados obtidos do modelo discreto do sistema tendo em conta o método de identificação implementado. A figura seguinte refere-se ao sinal de entrada do tipo PRBS aplicado ao atuador do sistema real.

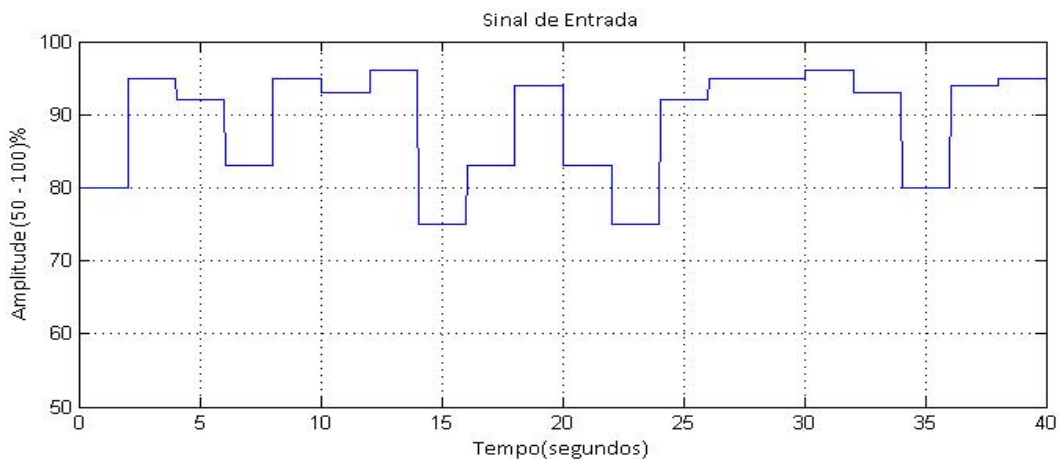


Figura 64 - Sinal de entrada do tipo PRBS para identificação dos parâmetros do sistema

As figuras seguintes mostram a evolução dos parâmetros identificados do sistema real para posterior obtenção do modelo da velocidade e o erro de estimação inerente à utilização do método *RLS* com fator de esquecimento direcional.

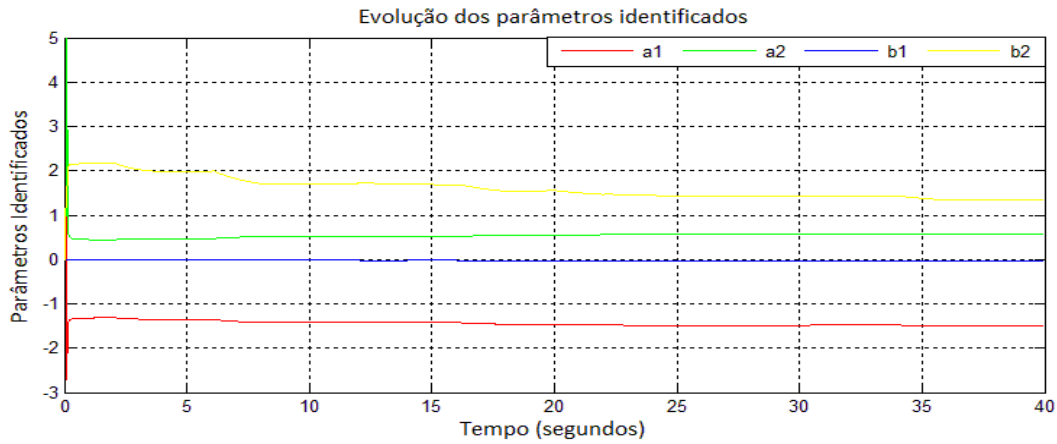


Figura 65 - Parâmetros identificados para obtenção do modelo discreto da velocidade

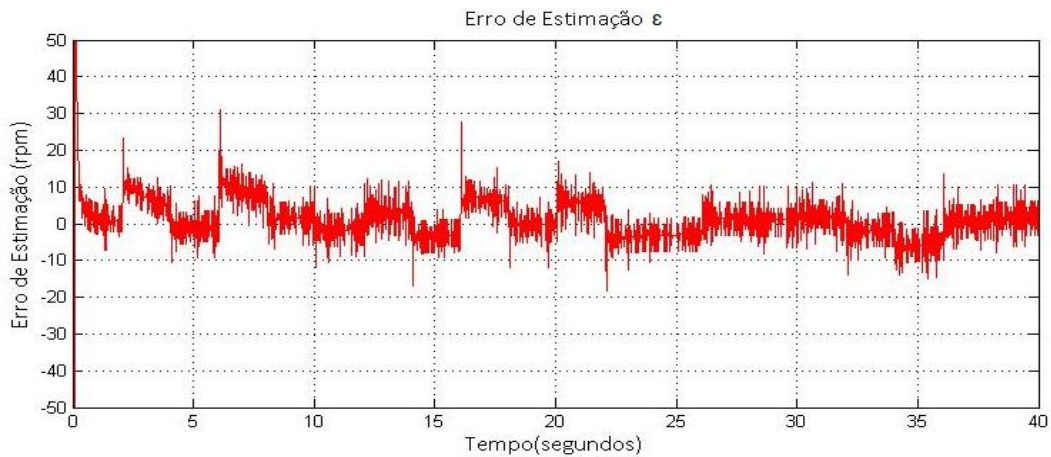


Figura 66 - Erro de estimação do modelo discreto da velocidade

Os parâmetros estimados utilizando o identificador de segunda ordem encontram-se na tabela seguinte.

$\hat{\theta}$	$\hat{\theta}_1$	$\hat{\theta}_2$	$\hat{\theta}_3$	$\hat{\theta}_4$
	a_1	a_2	b_1	b_2
Valores obtidos	-1.4943	0.5692	-0.0327	1.3528

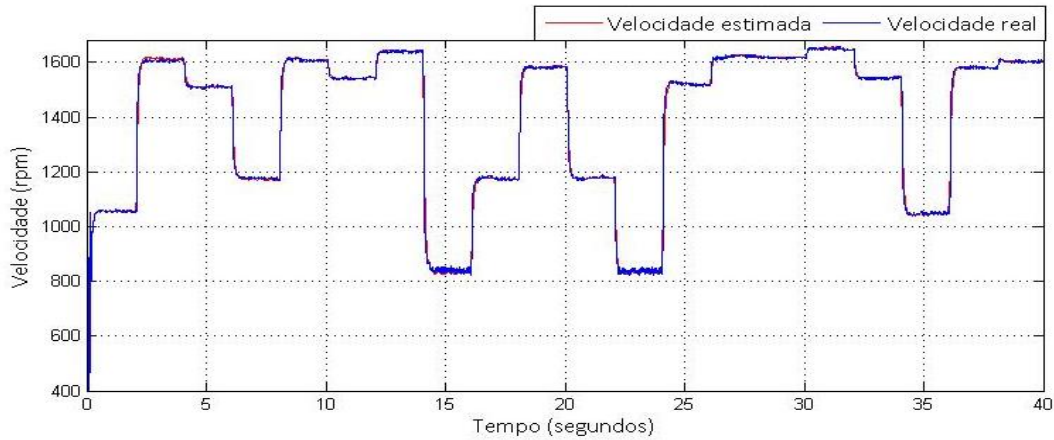
Tabela 4 - Parâmetros estimados do modelo discreto da velocidade

Com base nos resultados obtidos da tabela 4 e na equação 6.6, o modelo discreto do tipo ARX para a velocidade do sistema é dado na seguinte forma:

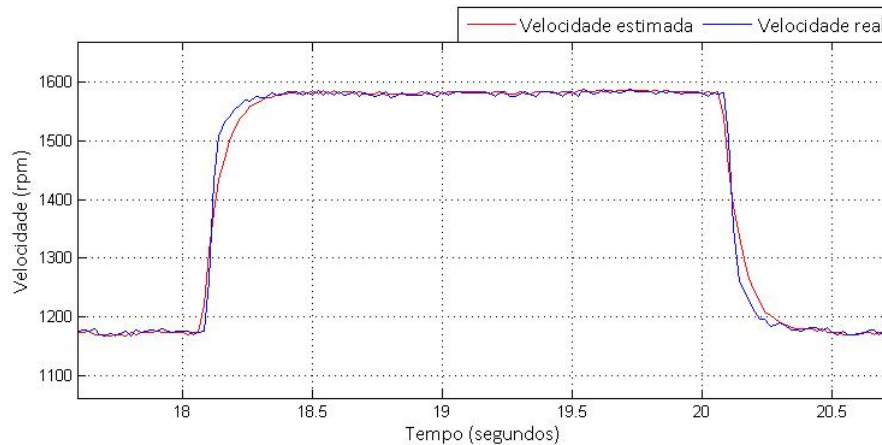
$$y_v(k) = 1.4943 y(k-1) - 0.5692 y(k-2) - 0.0327 u(k-1) + 1.3528 u(k-2) \quad (6.19)$$

em que y_v corresponde ao sinal de saída do modelo, ou seja, o sinal de velocidade. Os regressores da saída correspondem aos sinais de saída do sistema real, obtidos pela leitura

do codificador de quadratura. De acordo com o modelo discreto apresentado na forma de regressão linear em 6.19, apresenta-se de seguida o resultado da saída do sistema identificado e do sistema real em malha aberta.



a) Resposta da velocidade do sistema real e do sistema identificado em malha aberta



b) Sinais de saída (velocidade) ampliados

Figura 67 - Sinais de saída de velocidade do sistema real e do sistema identificado

O gráfico da Figura 67 a) mostra o resultado do sinal de saída (velocidade) resultante da identificação dos parâmetros pelo método recursivo *RLS* e também apresenta o resultado do sinal de saída do sistema real. Os dois sinais de saída encontram-se praticamente sobrepostos, não sendo visíveis as diferenças. Na Figura 67 b) apresentam-se os dois sinais de saída ampliados. De acordo com este resultado, verifica-se uma diferença mínima entre os dois sinais de saída. Pode-se concluir que os parâmetros identificados pelo método *RLS* são próximos dos parâmetros do sistema real.

Por observação da Figura 67 b) e considerando o instante de tempo a partir de $t = 18$ segundos em que o sinal de saída (velocidade) se encontra no flanco ascendente, verifica-se um tempo de subida aproximadamente de 200 ms. Sendo assim, o período de amostragem definido de 20ms é adequado uma vez que cumpre um dos critérios de seleção do período de amostragem, apresentado no tópico 3.7. Este intervalo de amostragem é imposto ao *kernel* de tempo real FreeRTOS. Por fim, o modelo discreto da velocidade na forma de espaço de estados é dado por:

$$\begin{bmatrix} x_1(k+1) \\ x_2(k+1) \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -0.5692 & 1.4943 \end{bmatrix} \begin{bmatrix} x_1(k) \\ x_2(k) \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u(k) \quad (6.20)$$

$$y_v(k) = [1.3528 \quad -0.0327] \begin{bmatrix} x_1(k) \\ x_2(k) \end{bmatrix} \quad (6.21)$$

O modelo discreto da velocidade apresentado anteriormente corresponde a um modelo de 2ª ordem com 4 parâmetros identificados pelo método *RLS* com fator de esquecimento direcional. As variáveis de estado x_1 e x_2 correspondem à velocidade e à corrente, respetivamente. Os pólos do sistema em malha aberta representado pelo modelo discreto das expressões 6.20 e 6.21 estão localizados em $0.747 \pm j0.1047$ no plano Z. De seguida apresenta-se o mapa de pólos e zeros deste sistema no plano Z.

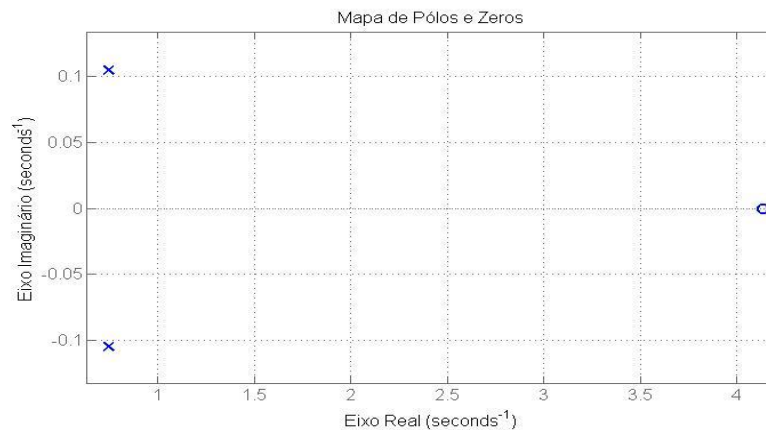


Figura 68 - Mapa de pólos e zeros do modelo discreto da velocidade

Observando a figura anterior, o modelo discreto para a velocidade apresenta dois pólos localizados dentro do círculo unitário e um zero fora do círculo unitário. A presença de um zero fora do círculo unitário poderá dever-se a um erro de medição provocado pela presença de ruído.

6.5.2 Identificação do modelo discreto da posição

Para a identificação do modelo discreto da posição do sistema real utilizou-se um identificador de segunda ordem do tipo *RLS* com fator de esquecimento direcional de 0.98. Geralmente, um modelo referente à posição é de terceira ordem, sendo este constituído por 6 parâmetros. Na presente dissertação, considera-se um modelo de segunda ordem para a posição, desprezando-se o pólo da componente elétrica do motor (valor da indutância da armadura muito pequeno). A interface gráfica (GUI) utilizada para visualizar os dados em tempo real relativos à identificação do sistema real encontra-se no Anexo A.1. Procedeu-se à identificação do modelo discreto da posição utilizando o mesmo período de amostragem do utilizado na identificação do modelo da velocidade. O sinal de entrada do tipo PRBS aplicado na entrada do sistema real para identificação do modelo discreto da posição é um sinal com valores percentuais compreendidos entre -90 e 90% da tensão aplicada ao motor DC.

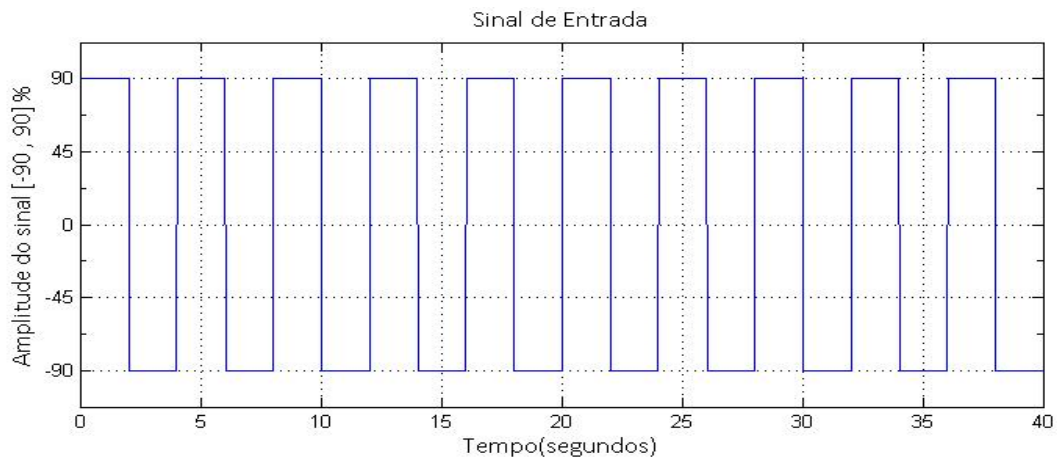


Figura 69 - Sinal de entrada do tipo PRBS para identificação do modelo discreto da posição

Os resultados da evolução dos parâmetros identificados bem como o do erro de estimação obtido pelo método *RLS* são apresentados de seguida.

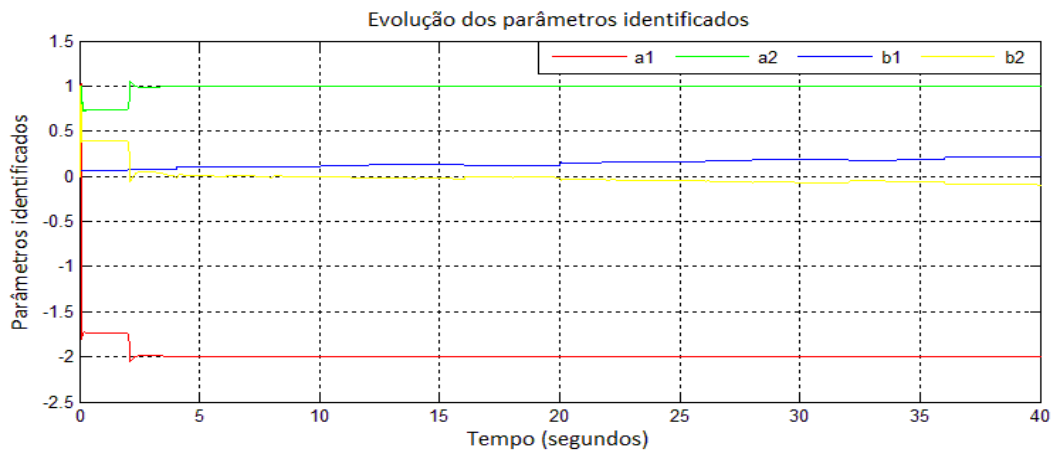


Figura 70 - Parâmetros identificados do modelo discreto da posição

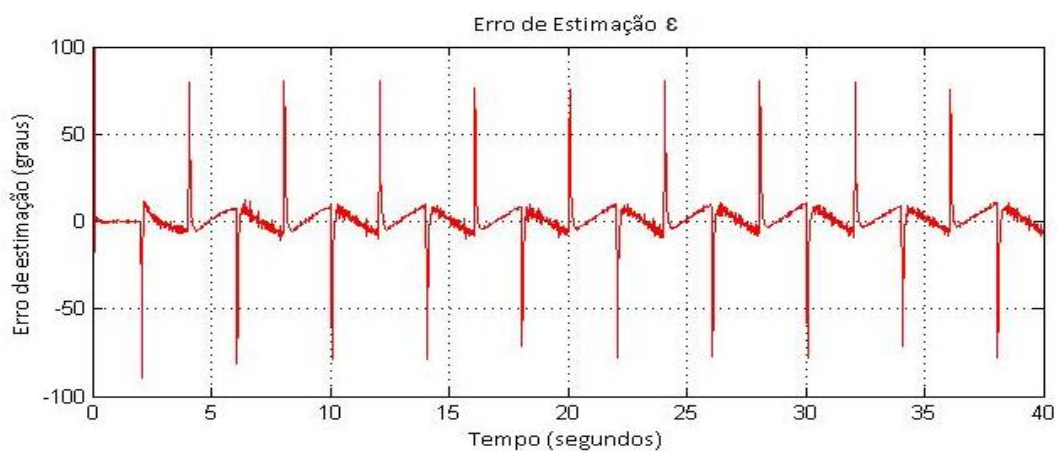


Figura 71 - Erro de estimação do modelo discreto da posição

Analisando de uma forma breve o resultado da Figura 70, nos instantes iniciais os 4 parâmetros do modelo do sistema apresentam uma oscilação, sendo que ao longo do tempo convergem para valores constantes. De acordo com o identificador de segunda ordem do tipo *RLS* com esquecimento direcional, os parâmetros identificados apresentam-se na tabela seguinte.

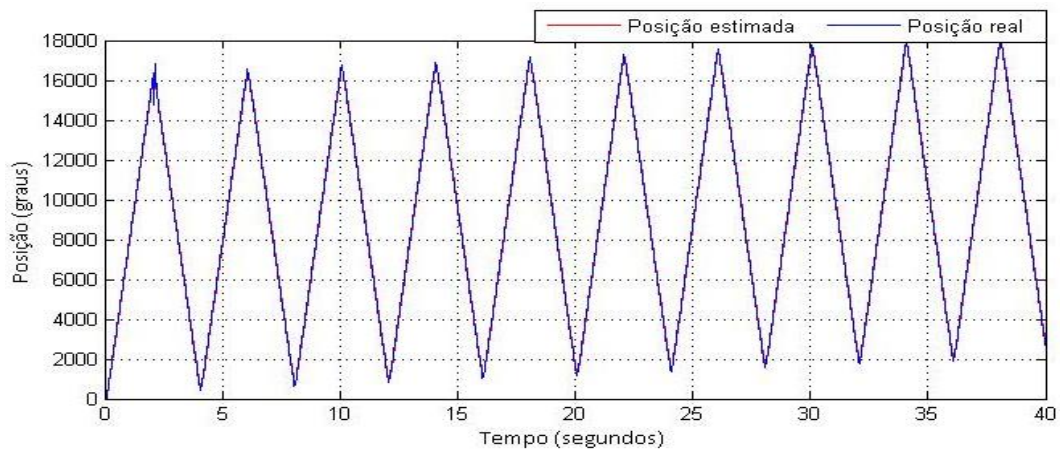
$\hat{\theta}$	$\hat{\theta}_1$	$\hat{\theta}_2$	$\hat{\theta}_3$	$\hat{\theta}_4$
	a_1	a_2	b_1	b_2
Valores obtidos	-1.9954	0.9964	0.2097	-0.0875

Tabela 5 - Parâmetros estimados do modelo discreto da posição

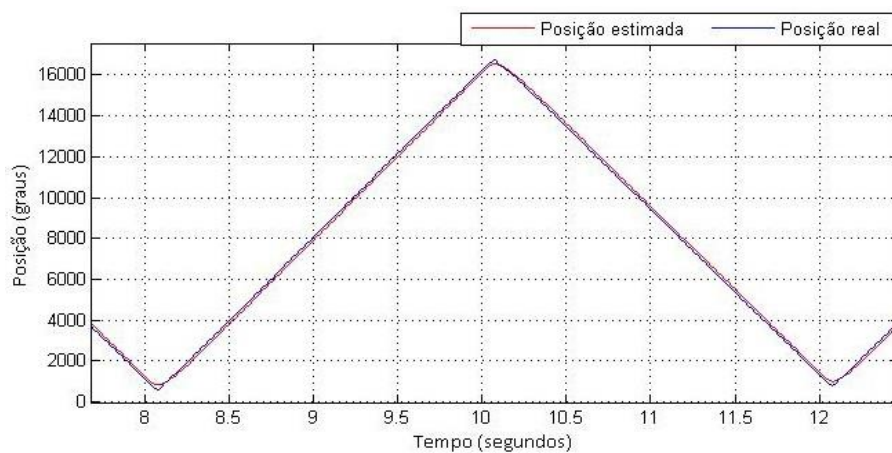
Considerando os parâmetros estimados da tabela 5, obtém-se o modelo discreto da posição por uma equação de regressão linear:

$$y_p(k) = 1.9954 y(k-1) - 0.9964 y(k-2) + 0.2097 u(k-1) - 0.0875 u(k-2) \quad (6.22)$$

Com base no modelo apresentado na expressão 6.22, apresenta-se de seguida o sinal de posição do sistema estimado. Também se apresenta o sinal de posição do sistema real, proveniente da leitura do codificador.



a) Resposta da posição do sistema real e do sistema identificado em malha aberta



b) Sinais de saída da posição ampliados

Figura 72 - Sinais da posição do sistema real e do sistema identificado

Para os sinais de posição do sistema real e do sistema identificado da Figura 72, verifica-se uma sobreposição dos dois sinais. Na Figura 72 b) observa-se com mais detalhe estes dois sinais. Com isto, o modelo estimado da posição aproxima-se do modelo do sistema real, uma vez que os sinais de saída apresentam um comportamento semelhante. Por fim, o modelo discreto da posição na forma de espaço de estados é obtido de forma análoga ao do modelo da velocidade representado nas expressões 6.20 e 6.21.

$$\begin{bmatrix} x_1(k+1) \\ x_2(k+1) \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -0.9964 & 1.9954 \end{bmatrix} \begin{bmatrix} x_1(k) \\ x_2(k) \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u(k) \quad (6.23)$$

$$y_p(k) = [-0.0875 \quad 0.2097] \begin{bmatrix} x_1(k) \\ x_2(k) \end{bmatrix} \quad (6.24)$$

O modelo discreto da posição do sistema na forma de espaço de estados representado pelas expressões 6.23 e 6.24 é de segunda ordem em que as variáveis de estado x_1 e x_2 correspondem à velocidade w e posição θ , respetivamente. Tratando-se de um modelo reduzido de 2ª ordem, são obtidos dois pólos e um zero no plano Z.

De seguida apresenta-se o mapa de pólos e zeros do sistema em malha aberta.

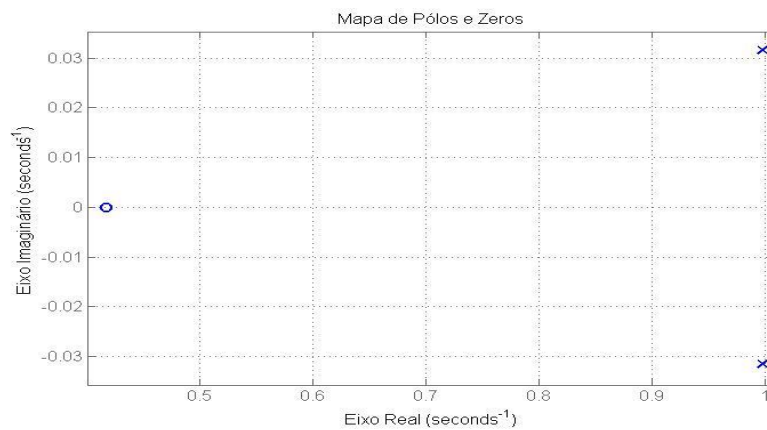


Figura 73 - Mapa de pólos e zeros do modelo discreto da posição

De acordo com o mapa de pólos e zeros apresentado na Figura 73, o sistema em malha aberta tem dois pólos com parte imaginária desprezável e um zero dentro do círculo unitário

Uma vez obtidos os modelos discretos do sistema real, cujos parâmetros foram estimados pelo método recursivo dos mínimos quadrados com esquecimento direcional, no capítulo seguinte implementa-se um controlador de velocidade e de posição para o sistema real.

Capítulo 7 Controlo do Sistema

Neste capítulo apresentam-se os resultados experimentais do controlo de velocidade e de posição do sistema real definido no capítulo 5. O controlador utilizado é um controlador digital por realimentação de estados sem e com ação integral, tendo sido apresentado anteriormente no capítulo 4. Este tipo de controlo foi abordado anteriormente no capítulo 4 quando foram efetuadas simulações de um sistema com parâmetros físicos conhecidos, modelado em Simulink e utilizando um controlador digital (ou discreto), implementado em MATLAB. No presente capítulo, implementa-se um controlador de um sistema com parâmetros desconhecidos que foram estimados no capítulo 6.

7.1 Resultados Experimentais

Nesta secção apresentam-se os resultados dos ensaios efetuados ao sistema real no controlo da velocidade e posição em duas vertentes: sem ação integral e com ação integral. O controlo do sistema real por realimentação de estados, como já visto anteriormente utiliza o modelo do sistema na forma de espaço de estados. Por fim, são apresentados os resultados dos ensaios ao sistema com variação de carga.

Para efetuar os ensaios ao sistema real no controlo da velocidade e da posição, considerou-se o período de amostragem selecionado no processo de identificação dos parâmetros do sistema real, ou seja, um período de amostragem de 20ms. O algoritmo de controlo do sistema real foi implementado em linguagem C sob o *kernel* de tempo real FreeRTOS. Como já referido, o fluxograma do processo de controlo do sistema motor-carga pode ser consultado no Anexo B.

Adotou-se a metodologia de cálculo dos ganhos de realimentação de estados via *Ackermann*, tendo sido implementado na linguagem de programação C. Desenvolveram-se duas interfaces gráficas (GUI) em MATLAB, uma para o controlo do sistema sem ação integral [Anexo A.2] e outra para o controlo com ação integral [Anexo A.3]. Estas interfaces permitem o cálculo dos ganhos de realimentação de estados pelo método de *Ackermann* utilizando os requisitos pretendidos para a malha fechada (tempo de estabelecimento e sobrelevação). Para além do cálculo dos ganhos, as interfaces permitem a visualização dos resultados dos ensaios efetuados no controlo da velocidade e da posição do sistema motor-carga.

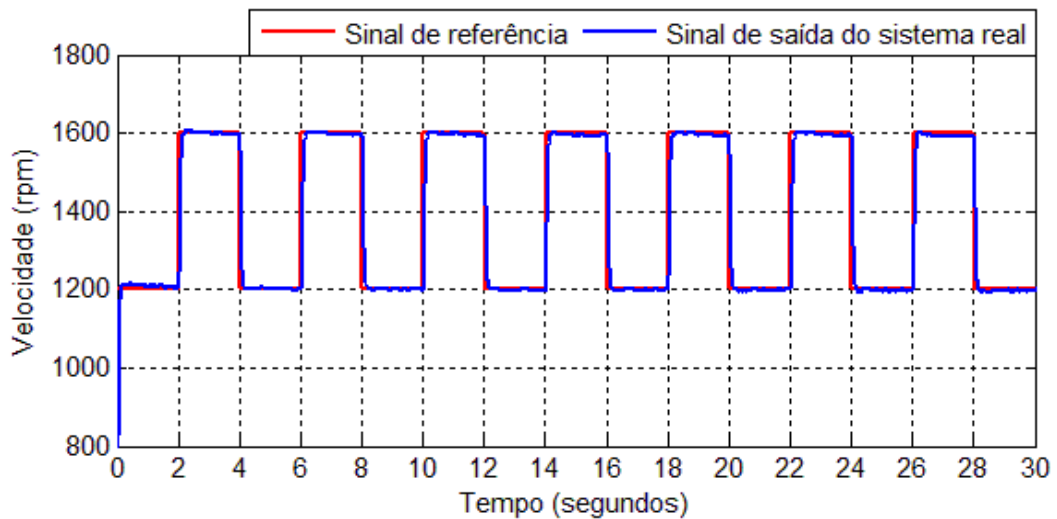
7.1.1 Controlo da Velocidade por Realimentação de Estados

Os resultados apresentados neste tópico referem-se aos ensaios efetuados com o controlador de velocidade por realimentação de estados, tendo sido feita a medição da velocidade e de corrente utilizando a eletrónica de medição descrita no capítulo 5. Como primeiro ensaio, utilizou-se um controlador de segunda ordem com a introdução de um ganho de pré-compensação na entrada de referência. Considerou-se o modelo discreto da velocidade na forma de espaço de estados obtido no capítulo 6 (tópico 6.5.1) para a obtenção dos ganhos de realimentação de estados pelo método de *Ackermann*. Definiu-se um tempo de estabelecimento de 0.65 segundos e uma sobrelevação máxima de 5% como requisitos de desempenho do sistema em malha fechada. A obtenção dos pólos na forma discreta do sistema em malha fechada bem como o cálculo dos ganhos de realimentação teve como base a

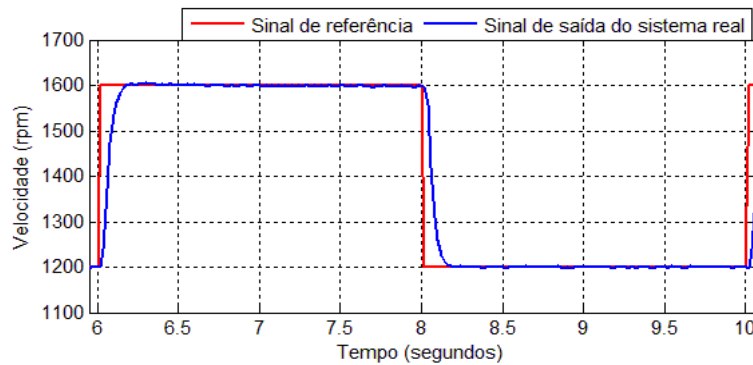
metodologia adotada anteriormente nos tópicos 4.5.2.2 e 4.5.3.1. Os pólos discretos do sistema em malha fechada encontram-se localizados no plano Z em $0.842 \pm j0.1423$. O sinal de controlo u utilizado no controlo da velocidade do sistema real foi definido na equação 4.36 e é dado pela seguinte expressão, tendo em conta os ganhos obtidos pelo método de Ackermann.

$$u(k) = r N - 0.213 w(k) + 0.259 i(k) \quad (7.1)$$

Na equação apresentada anteriormente, w representa a velocidade lida pelo codificador, i a corrente lida pelo sensor de corrente, r a velocidade de referência e N o ganho de compensação. Implementou-se um controlador de velocidade do sistema tendo em conta duas referências de velocidade, 1200 e 1600 rpm, sendo os valores atualizados cada 2 segundos. Para as referências consideradas, 1200 e 1600 rpm, utilizaram-se os ganhos de compensação na entrada de referência de 1.512 e 1.294, respetivamente. De salientar que os ganhos de compensação utilizados representam um fator de escala da entrada de referência. De seguida, apresentam-se os sinais envolvidos no controlo da velocidade do sistema real que resultam do ensaio efetuado.



a) Sinais de velocidade e de referência



b) Sinais de velocidade e de referência ampliados

Figura 74 - Sinais de saída (velocidade) e de referência do sistema real sem ação integral

Observando os sinais representados na Figura 74, verifica-se que a velocidade do sistema real segue as duas referências. De acordo com a Figura 74 b), a resposta do sistema apresenta um tempo de estabelecimento inferior a 0.5 segundos e uma sobrelevação inferior a 1%. No entanto, verificam-se pequenas oscilações no sinal de saída em estado estacionário que se traduz em pequenas variações na velocidade, inferior a 8 rpm. A figura seguinte apresenta o erro, expresso em rpm, da velocidade quando o sistema se encontra em estado estacionário.

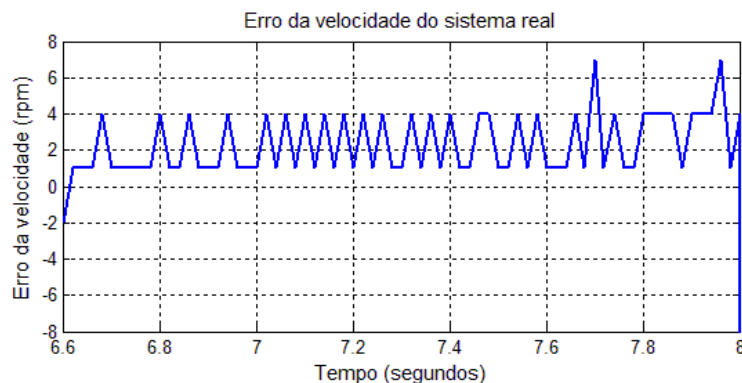


Figura 75 - Erro do sinal de saída no controlo de velocidade do sistema real sem ação integral

O erro representado na figura anterior corresponde à diferença entre a velocidade de referência e a velocidade medida pelo codificador. A existência de pequenas variações no valor de velocidade deve-se contudo a um erro de medição do codificador, tendo por isso pouca influência no sinal de controlo u .

De seguida, apresentam-se os sinais de controlo e da variável de estado corrente do sistema real.

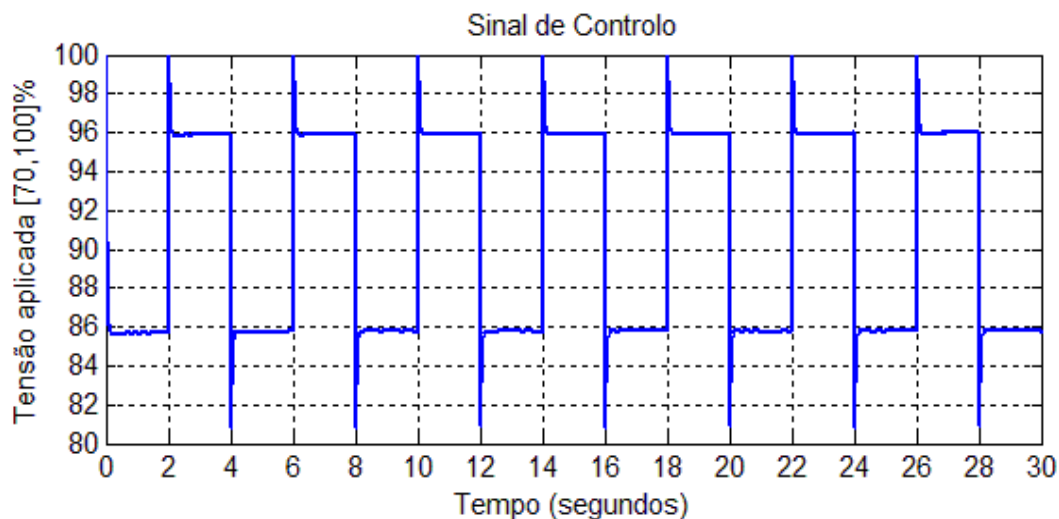


Figura 76 - Sinal de controlo da velocidade do sistema real sem ação integral

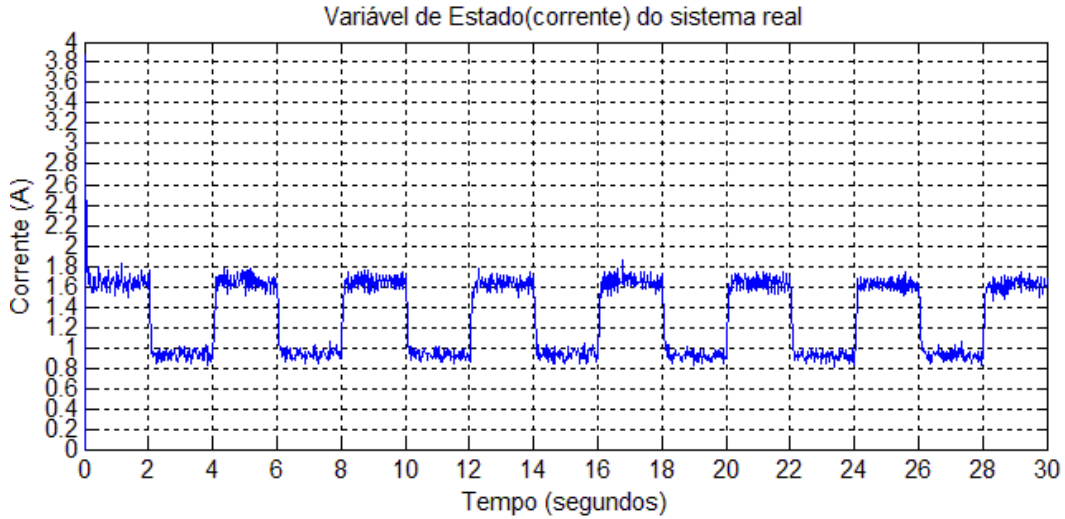


Figura 77 - Sinal da variável de estado corrente do sistema real sem ação integral

Analisando o sinal de controle (Figura 76) no estado estacionário, existe uma variação entre aproximadamente 86% e 96% da tensão aplicada ao sistema real. À velocidade de 1200 rpm a corrente medida pelo codificador é aproximadamente 1.7A e, por outro lado, para a velocidade de 1600 rpm a corrente é aproximadamente 0.9A. Verifica-se uma relação de proporcionalidade inversa entre o valor da corrente e o valor da velocidade, como referido na seção 2.2.1. Um aumento da velocidade do motor de controle provoca uma diminuição da corrente que percorre os enrolamentos do motor e, por sua vez, um aumento da carga. O comportamento descrito refere-se a situações em que o sistema não se encontra sob influência de uma carga externa. O sinal de corrente é afetado por ruído de medição, na ordem dos 0.1A. Contudo, este ruído não afeta o sinal de velocidade do sistema, no que respeita ao erro no estado estacionário e tem uma influência reduzida no cálculo do sinal de controle u . Conclui-se que a implementação de um controlador de velocidade por realimentação de estados utilizando um ganho de pré-compensação N não é um método prático no que se refere à utilização de um ganho escolhido por “tentativa e erro” como fator de escala da entrada de referência.

De seguida implementa-se o controlador de velocidade do sistema real com ação integral. Considera-se o modelo discreto da velocidade obtido pela identificação paramétrica do sistema real, dado pelas expressões 6.20 e 6.21. Adicionando o termo adicional ξ referente à ação integral, tem-se a seguinte representação discreta da velocidade por espaço de estados:

$$\begin{bmatrix} w(k+1) \\ i(k+1) \\ \varepsilon(k+1) \end{bmatrix} = \begin{bmatrix} A_{dv} & \mathbf{0} \\ -C_{dv} & 1 \end{bmatrix} \begin{bmatrix} w(k) \\ i(k) \\ \varepsilon(k) \end{bmatrix} + \begin{bmatrix} B_{dv} \\ 0 \end{bmatrix} u(k) \quad (7.2)$$

$$y_v(k) = [C_{dv} \ 0] \begin{bmatrix} w(k) \\ i(k) \end{bmatrix} \quad (7.3)$$

No modelo apresentado anteriormente, Substituindo as expressões 6.20 e 6.21 nas expressões 7.2 e 7.3, tem-se o modelo discreto para a velocidade com ação integral:

$$\begin{bmatrix} w(k+1) \\ i(k+1) \\ \varepsilon(k+1) \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ -0.569 & 1.494 & 0 \\ -1.352 & 0.032 & 1 \end{bmatrix} \begin{bmatrix} w(k) \\ i(k) \\ \varepsilon(k) \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} u(k) \quad (7.4)$$

$$y_v(k) = [1.352 \quad -0.032 \quad 0] \begin{bmatrix} w(k) \\ i(k) \\ \varepsilon(k) \end{bmatrix} \quad (7.5)$$

Para o cálculo dos ganhos de realimentação de estados, consideram-se os mesmos requisitos da resposta transitória do sistema real sem ação integral. O pólo adicional, referente ao termo da ação integral é posicionado no plano Z em 0.9 de modo a cancelar o efeito do zero localizado em $z=1$. O efeito do pólo adicional seria dominante se o sistema em malha aberta não tivesse um zero localizado em 1 no plano Z. Utilizando o método de *Ackermann* para o cálculo dos ganhos de realimentação de estados, obtém-se a seguinte matriz de realimentação, de acordo com a expressão 4.47:

$$\mathbf{K}_d = [0.12 \quad -0.139 \quad 0.127] \quad (7.6)$$

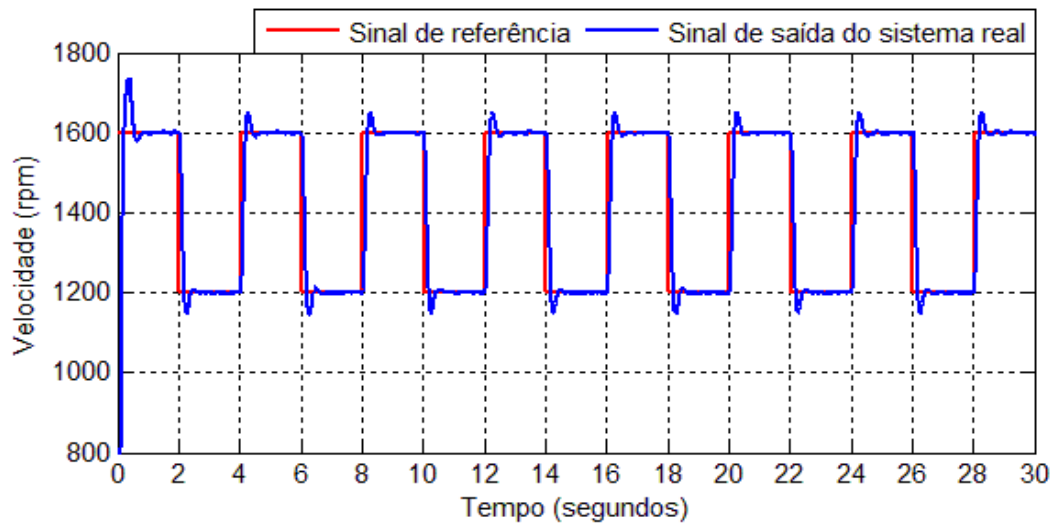
Utilizando os ganhos de realimentação de estados da expressão 7.6, o sinal de controlo para o sistema real com ação integral é dado na seguinte forma, tendo em conta a equação 4.54 apresentada anteriormente.

$$u(k) = k_{dint} (\varepsilon(k-1) + (r(k) - y(k))) - \mathbf{K}_{di} \mathbf{x}(k) \quad (7.7)$$

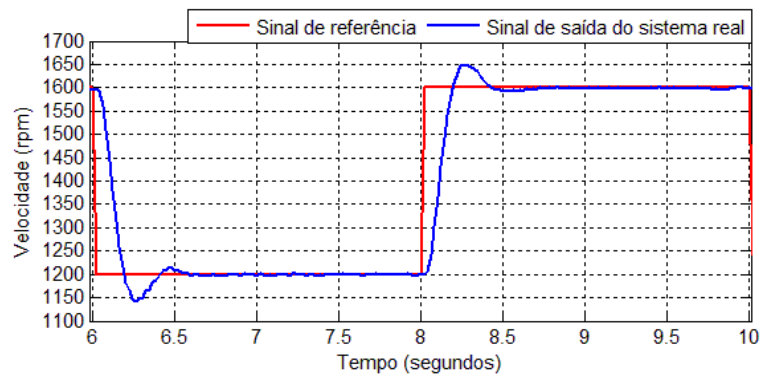
No controlo da velocidade do sistema real com ação integral, utiliza-se um modelo discreto de velocidade de terceira ordem, cujas variáveis de estado \mathbf{x} são dadas pela velocidade, corrente e pelo termo da ação integral. Simplificando a equação 7.7, obtém-se a equação simplificada do sinal de controlo u .

$$u(k) = 0.127 (\varepsilon(k-1) + (r(k) - y(k))) - 0.12 w(k) + 0.139 i(k) \quad (7.8)$$

De seguida são apresentados os resultados do comportamento da dinâmica do sistema real para os requisitos da resposta transitória, isto é, um tempo de estabelecimento de 0.65 segundos e uma sobrelevação máxima de 5%, também considerados no caso do sistema sem ação integral.



a) Sinais de velocidade e de referência



b) Sinais de velocidade e de referência ampliados

Figura 78 - Sinais de saída (velocidade) e de referência do sistema real com ação integral

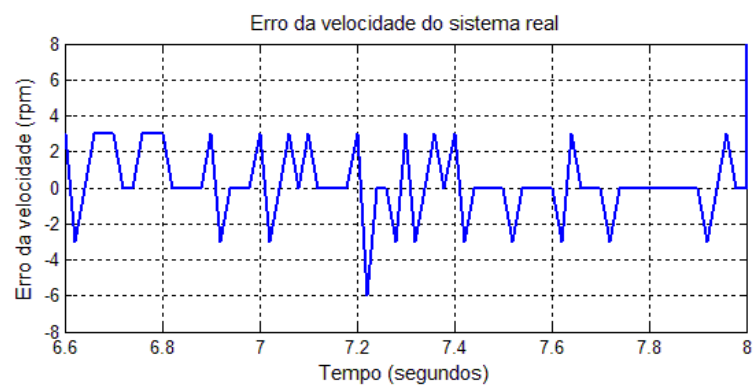


Figura 79 - Erro do sinal de saída no controle de velocidade do sistema real com ação integral

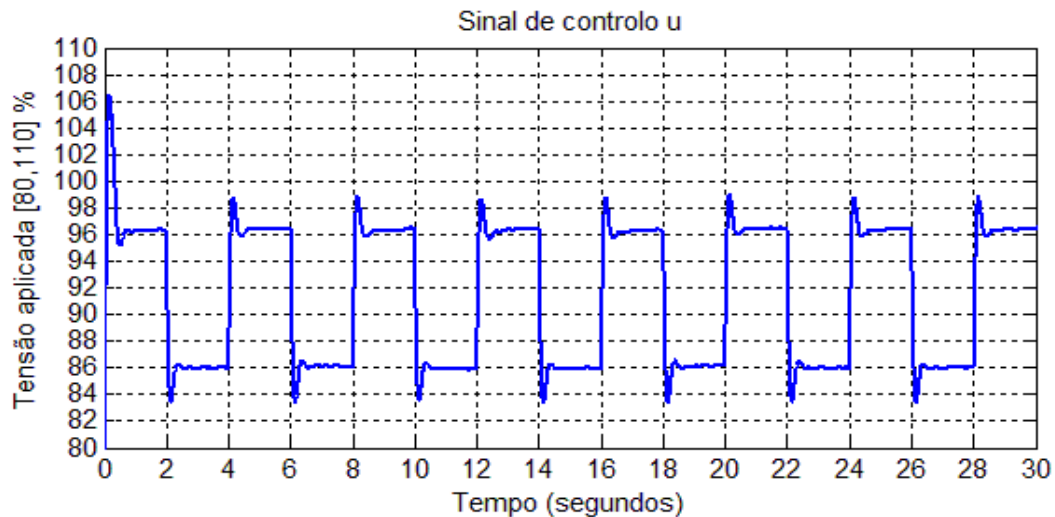


Figura 80 - Sinal de controle da velocidade do sistema real com ação integral

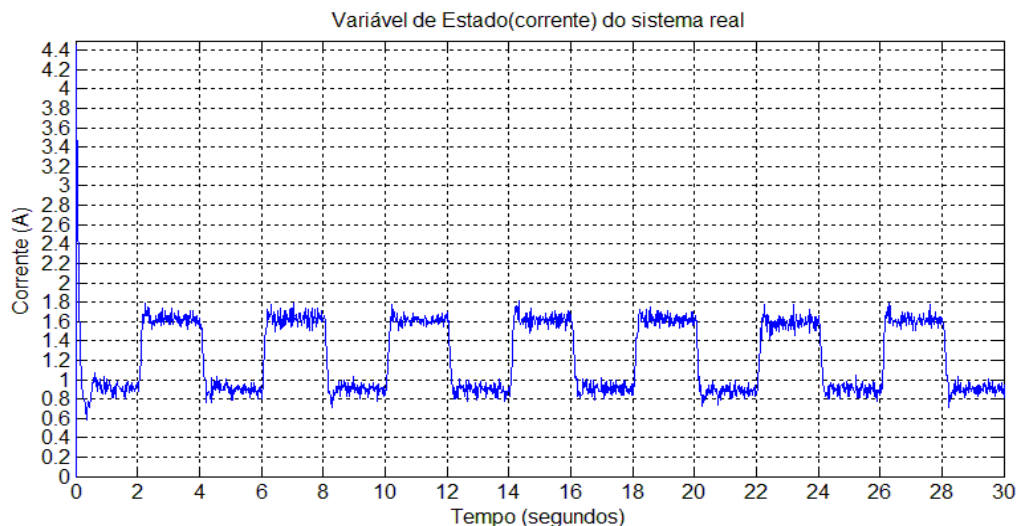


Figura 81 - Sinal da variável de estado corrente do sistema real com ação integral

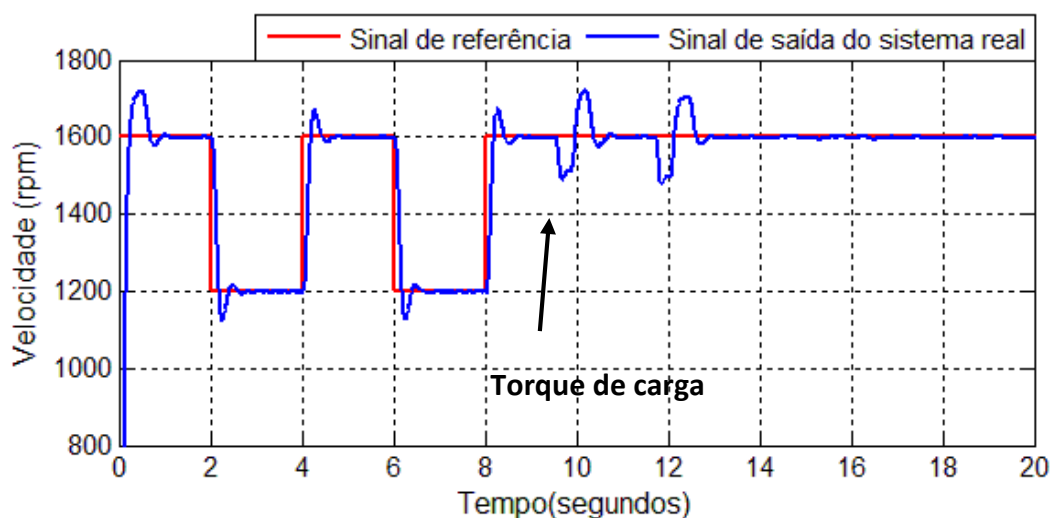
Analisando os resultados da Figura 78 relativamente ao controlo da velocidade do sistema com ação integral, verifica-se que os requisitos para a resposta transitória são cumpridos. Para as duas referências de velocidade, o sistema apresenta na sua resposta transitória uma sobrelevação aproximadamente de 3% e um tempo de estabelecimento aproximadamente de 0.5 segundos, podendo ser verificado na Figura 78 b). Comparativamente aos resultados obtidos para o sistema sem ação integral, a resposta transitória do sistema com ação integral é mais lenta. Contudo, a presença de ação integral no sistema real não compromete os requisitos considerados. Analisando de uma forma breve a Figura 79, verifica-se um erro de velocidade no estado estacionário inferior a 6 rpm. Isto permite concluir que houve uma melhoria do erro de velocidade comparativamente ao sistema sem ação integral.

Relativamente ao sinal de controlo (Figura 80) e da variável de estado corrente (Figura 81), apresentam um comportamento em estado estacionário semelhante ao do sistema sem ação integral. Em estado transitório, tanto o sinal de controlo como o de corrente apresentam

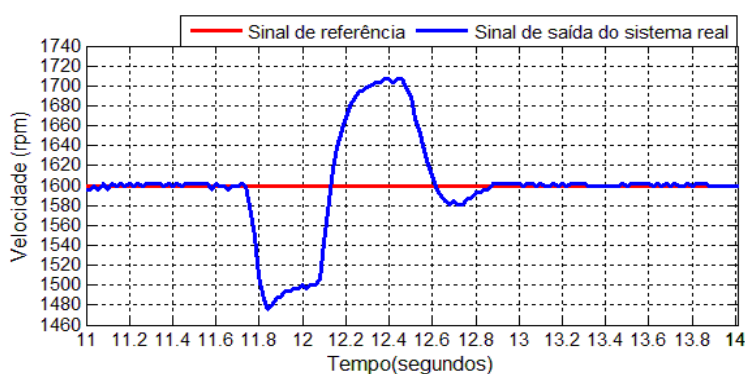
sobrelevação e um tempo de estabelecimento superior ao do sistema sem ação integral, tornando a resposta do sistema mais lenta.

7.1.1.1 Introdução de um torque de carga externo

Como visto anteriormente, o controlador de velocidade do sistema real com ação integral é uma boa alternativa ao controlador de velocidade sem ação integral utilizando o ganho de compensação N. Nesta seção é analisado o comportamento do sistema quando se impõe variação de torque através do motor de carga que produz uma dada corrente elétrica. Conforme referido no capítulo 5, o segundo motor é utilizado como carga e é utilizada a fonte de corrente constante referenciada no tópico 5.3.3. De seguida apresentam-se os resultados dos sinais envolvidos no controlo da velocidade do sistema real com ação integral e com a influência de um torque de carga que corresponde a uma corrente elétrica de 250mA.



a) Sinais de velocidade e de referência



b) Sinais de velocidade e de referência ampliados

Figura 82 - Influência de um torque carga externo no controlo da velocidade do sistema real

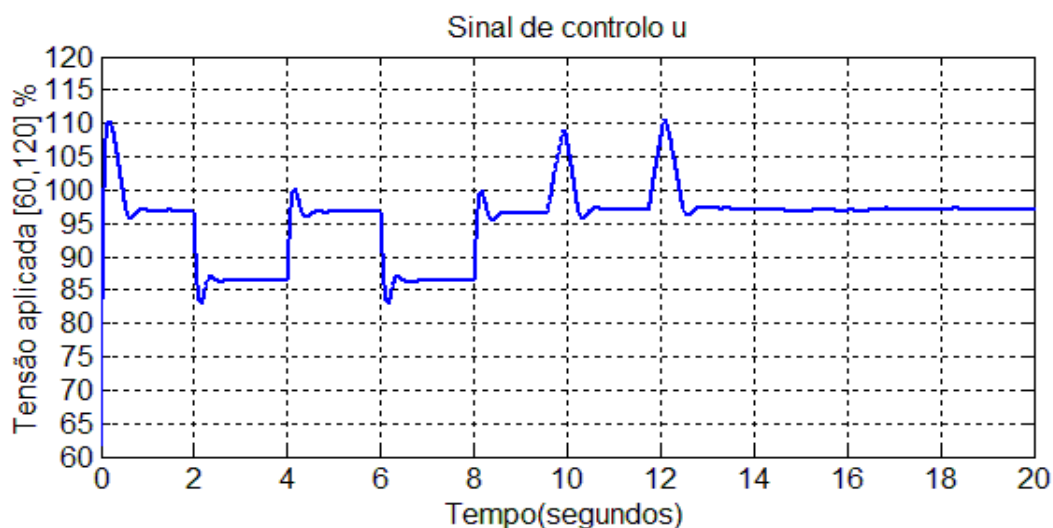
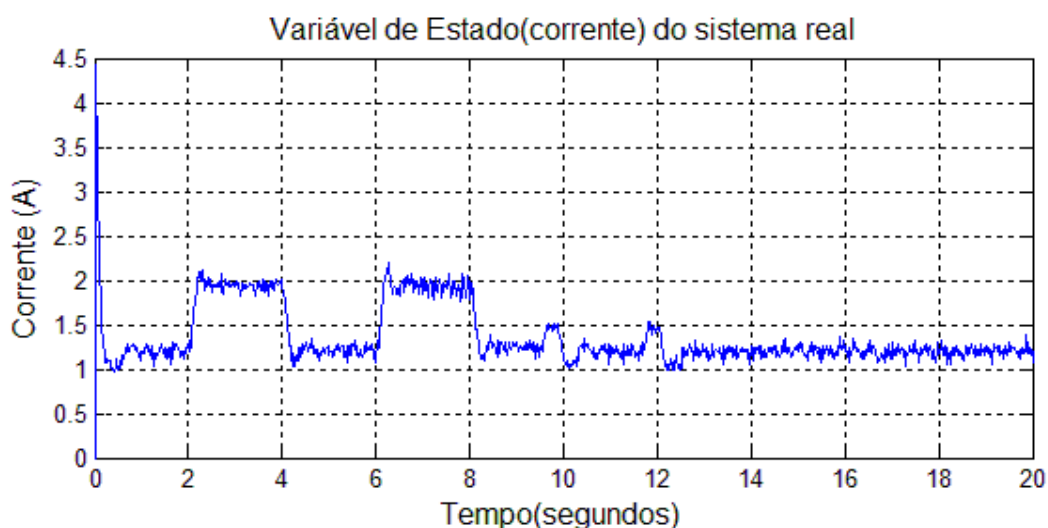


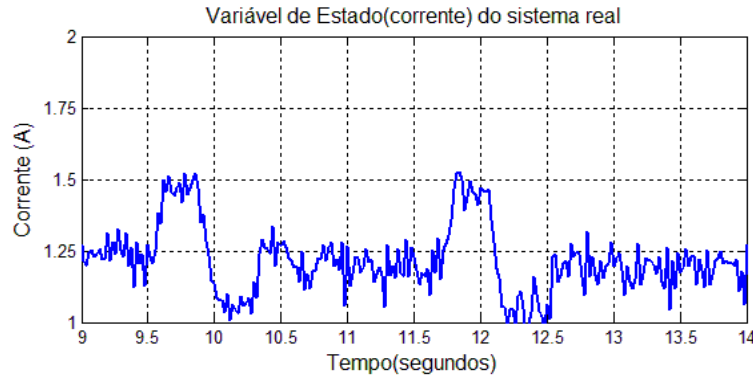
Figura 83 - Sinal de controlo da velocidade do sistema real com influência de uma carga

A Figura 82 apresenta o comportamento da velocidade do sistema quando lhe é introduzida um torque carga no instante de tempo $t = 11.7$ segundos. Com a variação de carga no sistema, a velocidade apresenta uma diminuição na ordem dos 100 rpm, demorando aproximadamente 1.3 segundos a alcançar o estado estacionário. No ensaio efetuado, a introdução de um torque de carga no sistema de controlo quando este se encontra no estado estacionário faz com que a tensão aplicada aumente na ordem dos 12%. Em regime estacionário, a velocidade apresenta um erro inferior a 10 rpm, sendo este erro associado a um erro de medição do codificador, já referido anteriormente.

Os resultados apresentados de seguida referem-se ao comportamento da variável de estado corrente.



a) Sinal da corrente do sistema real com a influência de uma carga de 250mA



b) Introdução de uma carga de 250mA no sistema real

Figura 84 - Sinal da variável de estado corrente do sistema real com influência de uma carga

O comportamento da variável de estado corrente nos instantes de tempo t e t_1 é válido na medida em que o valor da corrente consumida pelo motor de controlo é aproximadamente 250mA. No entanto, o valor de corrente no estado estacionário apresenta um erro de cerca de 0.2 A, estando associado a um erro de medição do sensor. De acordo com os resultados obtidos neste tópico, conclui-se que o sistema com ação integral, quando sujeito a uma variação de carga, atinge novamente o estado estacionário com um tempo de estabelecimento e uma sobrelevação acima dos valores definidos para os requisitos da resposta transitória. No entanto, verifica-se uma diminuição do erro do sinal de saída (velocidade) do sistema no estado estacionário. Isto deve-se à ação integral introduzida no sistema, que tem um papel preponderante na diminuição do erro em regime estacionário.

7.1.2 Controlo da Posição por Realimentação de Estados

A metodologia utilizada na implementação de um controlador de posição por realimentação de estados é a mesma da utilizada no tópico anterior, referente ao controlador de velocidade. Como visto na seção 7.1.1, a utilização de um ganho de compensação no controlo do sistema por realimentação de estados permite corrigir o erro no estado estacionário, no entanto a escolha do ganho por tentativa e erro torna-se uma desvantagem. Com isto, implementou-se apenas um controlador de posição com ação integral de terceira ordem. O modelo discreto da posição obtido na seção 6.5.2 pode ser representado na seguinte forma:

$$\begin{bmatrix} \theta(k+1) \\ w(k+1) \end{bmatrix} = \begin{bmatrix} 1.9954 & -0.9964 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} \theta(k) \\ w(k) \end{bmatrix} + \begin{bmatrix} 1 \\ 0 \end{bmatrix} u(k) \quad (7.9)$$

$$y_p(k) = [0.2097 \quad -0.0875] \begin{bmatrix} \theta(k) \\ w(k) \end{bmatrix} \quad (7.10)$$

Considerando o modelo apresentado anteriormente e adicionando o termo da ação integral, obteve-se a seguinte representação discreta em espaço de estados com ação integral:

$$\begin{bmatrix} \theta(k+1) \\ w(k+1) \\ \varepsilon(k+1) \end{bmatrix} = \begin{bmatrix} \mathbf{A}_{dp} & 0 \\ -\mathbf{C}_{dp} & 1 \end{bmatrix} \begin{bmatrix} \theta(k) \\ w(k) \\ \varepsilon(k) \end{bmatrix} + \begin{bmatrix} \mathbf{B}_{dp} \\ 0 \end{bmatrix} u(k) \quad (7.11)$$

$$y_p(k) = [\mathbf{C}_{dp} \ 0] \begin{bmatrix} \theta(k) \\ w(k) \\ \varepsilon(k) \end{bmatrix} \quad (7.12)$$

$$\mathbf{A}_{dp} = \begin{bmatrix} 1.9954 & -0.9964 \\ 1 & 0 \end{bmatrix} \quad (7.13)$$

$$\mathbf{B}_{dp} = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \quad (7.14)$$

$$\mathbf{C}_{dp} = [0.2097 \quad -0.0875] \quad (7.15)$$

Substituindo as matrizes \mathbf{A}_{dp} , \mathbf{B}_{dp} e \mathbf{C}_{dp} na representação discreta da posição com ação integral,

$$\begin{bmatrix} \theta(k+1) \\ w(k+1) \\ \varepsilon(k+1) \end{bmatrix} = \begin{bmatrix} 1.9954 & -0.9964 & 0 \\ 1 & 0 & 0 \\ -0.209 & 0.087 & 1 \end{bmatrix} \begin{bmatrix} \theta(k) \\ w(k) \\ \varepsilon(k) \end{bmatrix} + \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} u(k) \quad (7.16)$$

$$y_p(k) = [0.209 \quad -0.087 \quad 0] \begin{bmatrix} \theta(k) \\ w(k) \\ \varepsilon(k) \end{bmatrix} \quad (7.17)$$

O modelo discreto da posição do sistema real apresentado anteriormente tem como variáveis de estado a posição θ , a velocidade w e o termo da ação integral ξ . No primeiro ensaio efetuado com o controlador de posição, definiram-se como requisitos da resposta transitória um tempo de estabelecimento de 0.2 segundos e uma sobrelevação máxima de 20%. Os pólos de malha fechada obtidos através da definição dos requisitos da resposta transitória encontram-se localizados em $0.296 \pm j0.506$. O pólo adicional, referente à ação integral é posicionado em $p_z = 0.94$ de modo a minimizar o efeito do zero localizado em $z = 1$. De forma análoga ao controlo da velocidade do sistema, calcularam-se os ganhos de realimentação de estados pelo método de *Ackermann*, tendo sido também utilizado no capítulo 4.

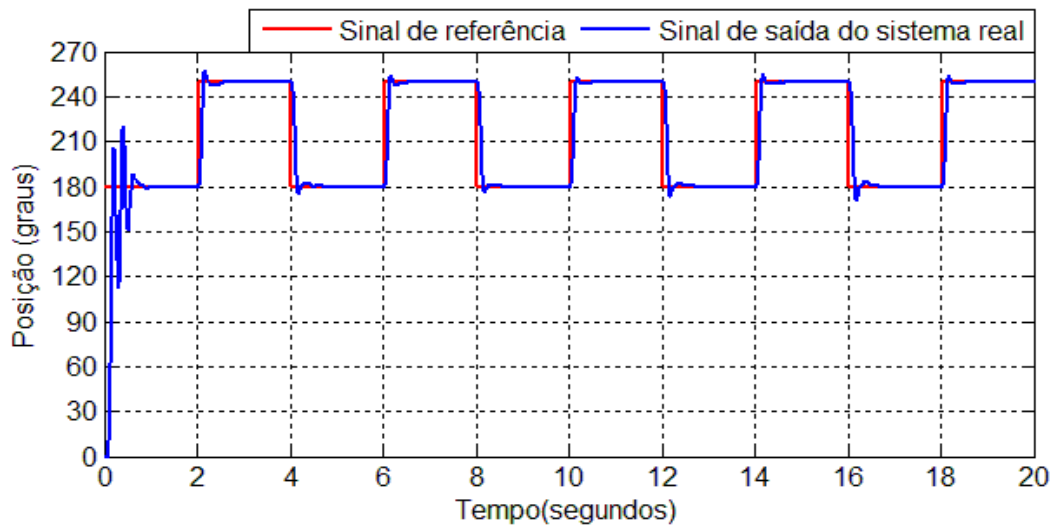
A matriz de realimentação de estados \mathbf{K}_{dp} com os ganhos obtidos pelo método de *Ackermann* é apresentada na seguinte expressão:

$$\mathbf{K}_{dp} = [1.165 \quad -0.613 \quad -0.248] \quad (7.18)$$

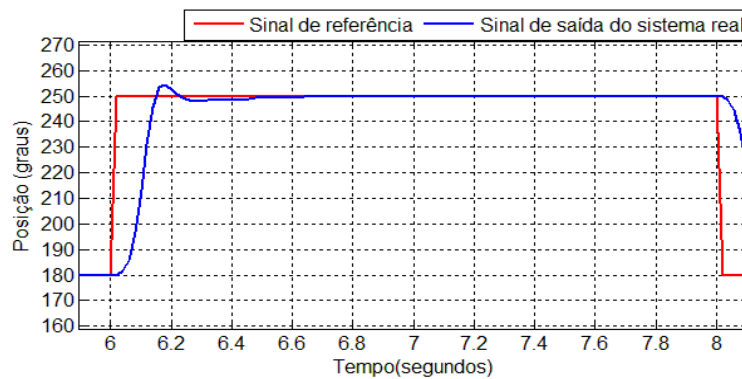
Tendo em conta a expressão 7.7, o sinal de controlo da posição do sistema real é dado por:

$$u(k) = 0.248 \left(\varepsilon(k-1) + (r(k) - y(k)) \right) + 0.613 w(k) - 1.165 \theta(k) \quad (7.19)$$

De seguida são apresentados os resultados dos sinais envolvidos no controlo do sistema real para duas referências de posição, 180 e 250 graus.

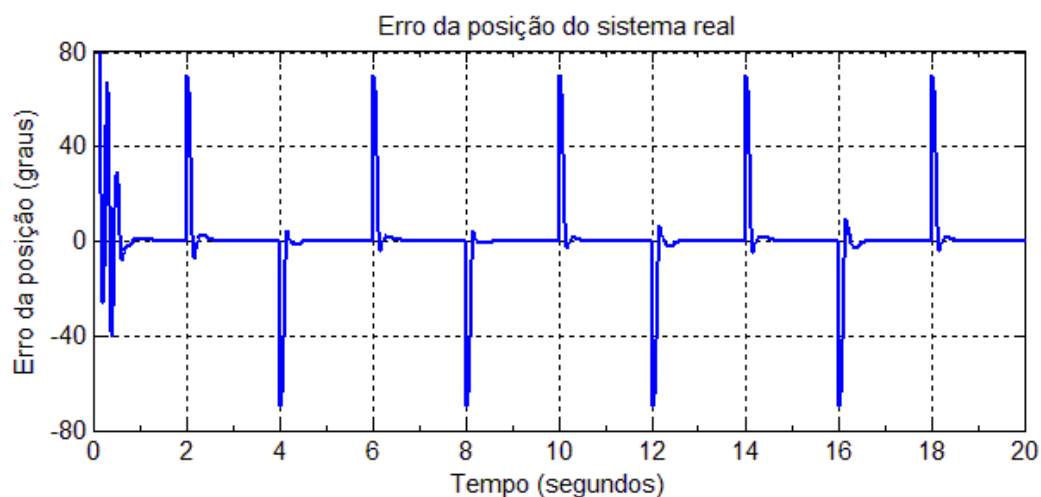


a) Sinais de posição e de referência

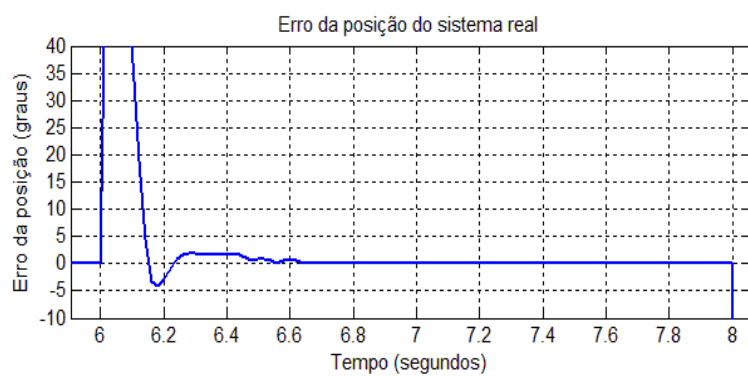


b) Sinais de posição e de referência ampliados

Figura 85 - Sinais de saída (posição) e de referência do sistema real com ação integral (Ensaio 1)



a) Sinal do erro de posição



b) Sinal ampliado

Figura 86 - Sinais do erro de posição do sistema real com ação integral (Ensaio 1)

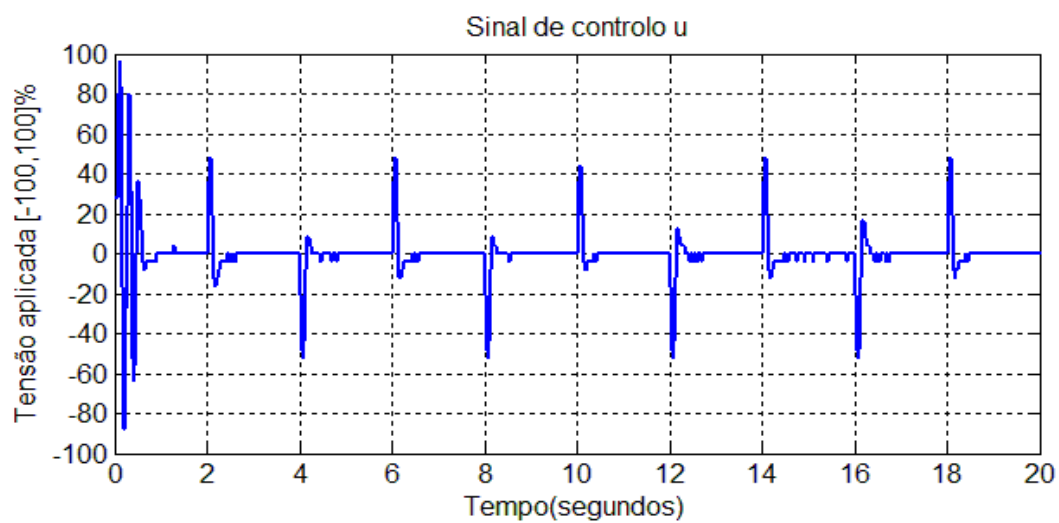


Figura 87 - Sinal de controlo da posição do sistema real com ação integral (Ensaio 1)

Por análise dos resultados apresentados anteriormente, a posição do sistema real segue a referência cumprindo os requisitos definidos para a resposta transitória. Na Figura 85 b), em regime transitório, o sistema apresenta um tempo de estabelecimento inferior a 0.2 segundos e uma sobrelevação na ordem dos 2%. Analisando o estado estacionário, o sistema não apresenta erro, sendo portanto, o estado em que a posição corresponde à posição de referência. Analisando a Figura 86, no intervalo de tempo $t = [6.6, 8]$ segundos que correspondente ao estado estacionário, o erro de posição é nulo.

No estado inicial, o sistema apresenta um comportamento instável, como se pode verificar nas figuras anteriores. Isto deve-se ao fato de o sinal de controlo u (Figura 87) se encontrar limitado entre $[-90, 90]$ % e o elevado binário que se verifica no arranque do motor DC, provocando uma instabilidade no sinal de posição. Relativamente ao sinal de controlo u , em regime transitório, apresenta um valor percentual de pico que varia entre $[-50, 50]$ %, dependendo da polaridade do sinal. No estado estacionário o sistema segue a posição de referência, em que o valor do sinal de controlo u é nulo.

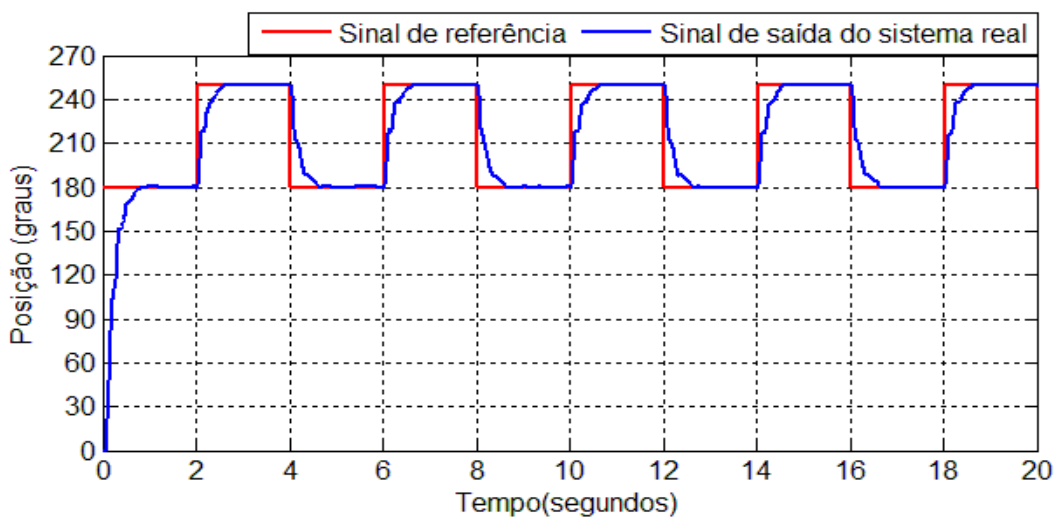
No segundo ensaio efetuado com o controlador de posição, posicionaram-se os dois pólos dominantes no plano Z em -0.2 e 0.2. O pólo do integrador foi posicionado no plano Z em 0.97 de modo a minimizar o efeito do zero localizado em $z=1$. Utilizando o método de Ackermann obtiveram-se os ganhos de realimentação de estados representados na seguinte matriz:

$$K_{dp} = [2.02 \quad -1.05 \quad | \quad -0.23] \quad (7.20)$$

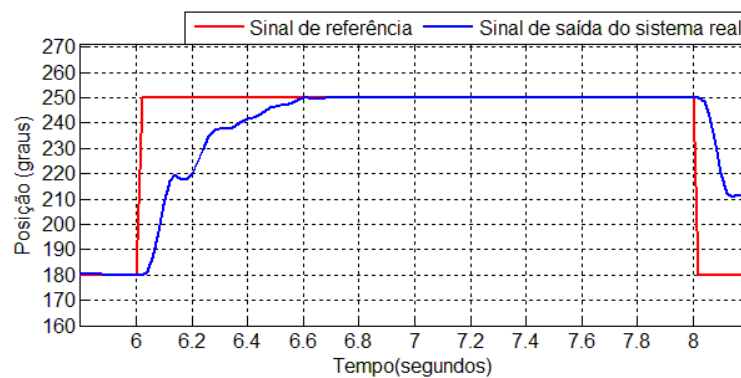
O sinal de controlo u com os respetivos ganhos de realimentação de estados da expressão 7.20 é dado por:

$$u(k) = 0.23 \left(\varepsilon(k-1) + (r(k) - y(k)) \right) + 1.05 w(k) - 2.02 \theta(k) \quad (7.21)$$

De seguida são apresentados os resultados obtidos considerando o mesmo tempo de amostragem e as mesmas referências de posição.

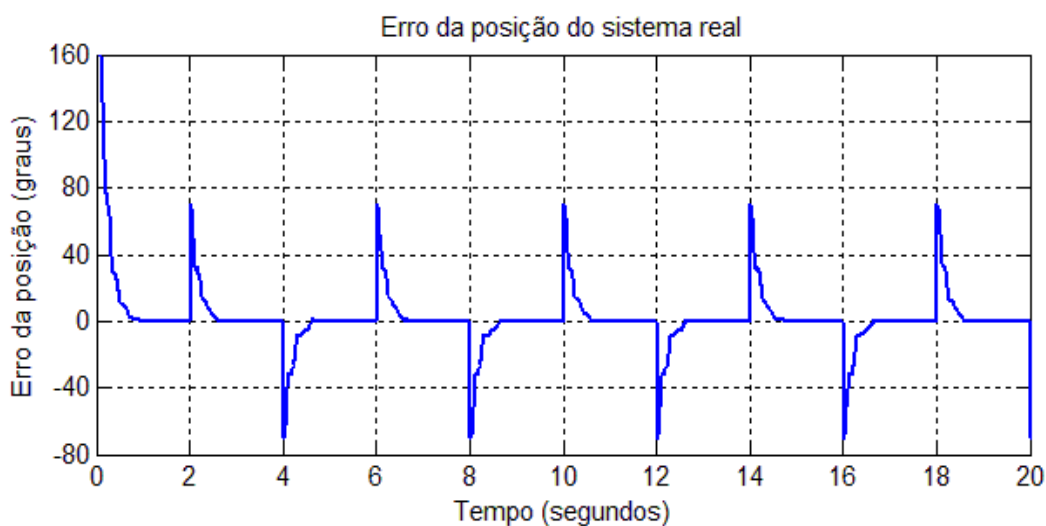


a) Sinais de posição e de referência

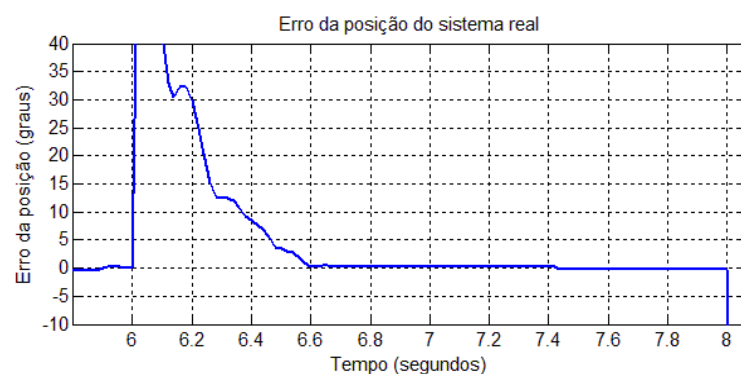


b) Sinais de posição e de referência ampliados

Figura 88 - Sinais de saída (posição) e de referência do sistema real com ação integral (Ensaio 2)



a) Sinal do erro de posição



b) Sinal ampliado

Figura 89 - Sinais do erro de posição do sistema real com ação integral (Ensaio 2)

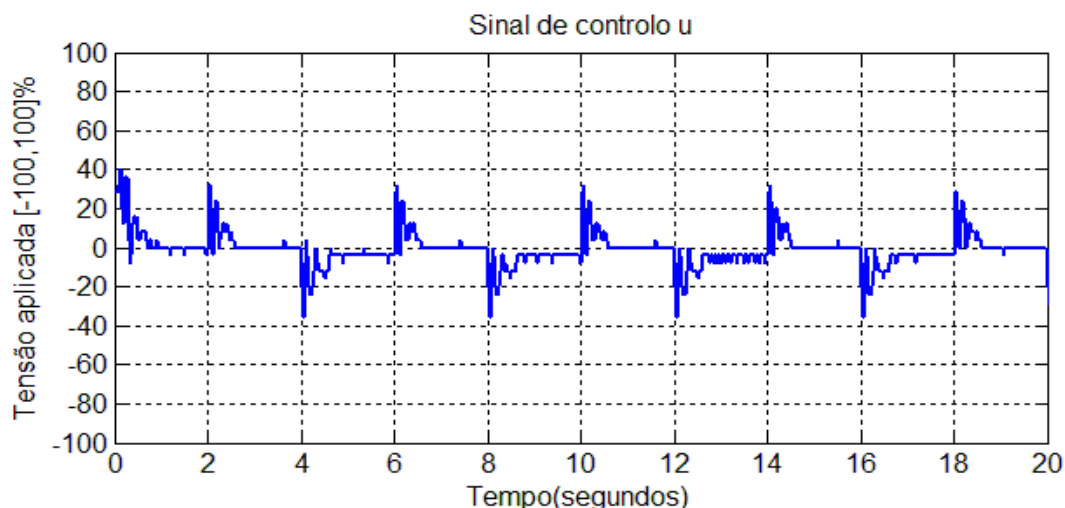


Figura 90 - Sinal de controle da posição do sistema real com ação integral (Ensaio 2)

Analizando os resultados da Figura 88, a resposta transitória do sinal de posição apresenta um tempo de estabelecimento aproximadamente de 0.5 segundos e não apresenta sobrelevação. Verifica-se em regime transitório um comportamento oscilatório do sinal de posição que decresce em amplitude até ser alcançado o regime estacionário (Figura 88 b)).

Analizando o resultado da Figura 89 b), o erro entre a posição de referência e a posição real (ou medida pelo codificador de quadratura) em regime estacionário é inferior a 0.5 graus, sendo um valor próximo de zero. De salientar que o regime estacionário representado na Figura 89 b) refere-se ao intervalo de tempo $t = [6.6, 8]$ segundos. O sinal de controle u , representado na Figura 90, apresenta um valor percentual de pico que varia aproximadamente entre $[-40, 40]$ %.

Comparando os dois ensaios efetuados, a resposta transitória do sistema no ensaio 1 é mais rápida em comparação com a resposta no ensaio 2, bem como a existência de sobrelevação no primeiro caso. De seguida, é apresentada uma tabela comparativa entre os dois ensaios.

	Tempo de Estabelecimento (seg)	Sobrelevação (%)	Erro no Estado estacionário (graus)	Dinâmica
Ensaio 1	0.2	2	Nulo	Rápida
Ensaio 2	0.5	0	0.5	Lenta

Tabela 6 - Comparação dos ensaios efetuados ao sistema real no controle de posição

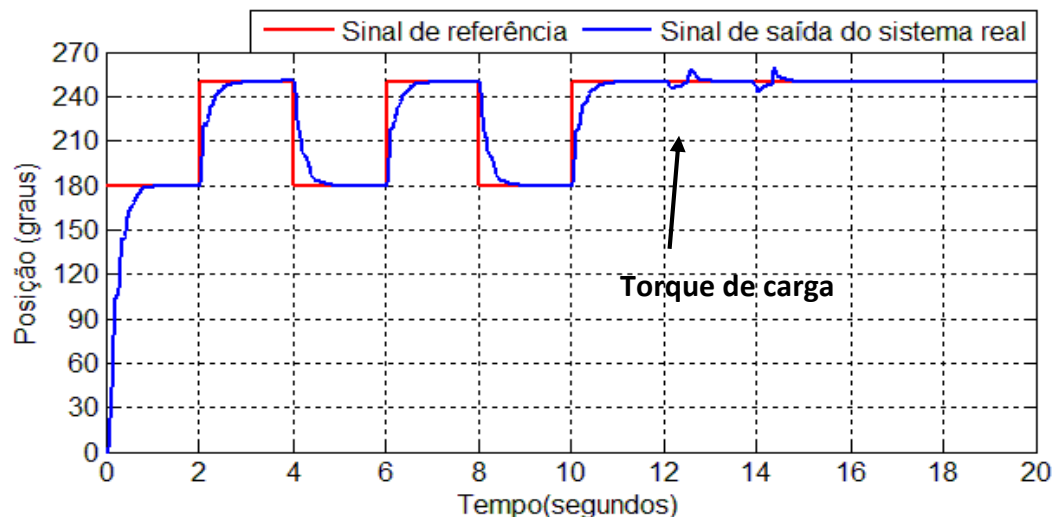
Os resultados apresentados na tabela 6 relativamente aos critérios de desempenho da resposta transitória do sistema real referem-se a um sistema de segunda ordem. Uma vez que se trata de um sistema de terceira ordem, os dois pólos (dominantes) definidos para a malha fechada estão diretamente relacionados com o fator de amortecimento e a frequência natural. A localização dos pólos dominantes em malha fechada permite caracterizar a resposta do sistema no que refere ao comportamento oscilatório.

Por observação dos resultados da tabela 6, pode-se concluir que no ensaio 1 a resposta do sistema apresenta características de um sistema sub-amortecido. Isto deve-se ao fato de os pólos de malha fechada para o sistema de segunda ordem terem componente real e imaginária, bem como o fator de amortecimento compreendido entre 0 e 1.

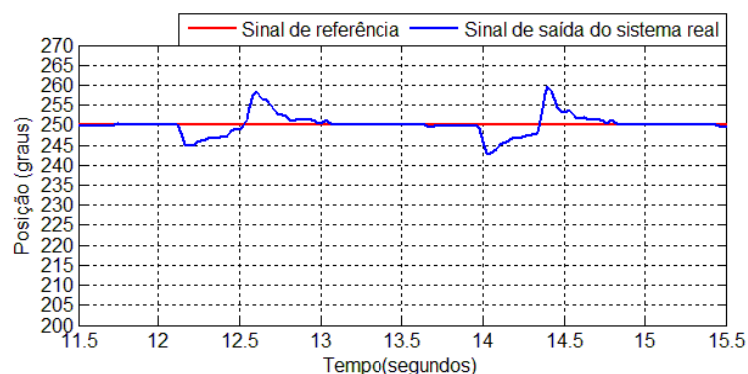
No ensaio 2, o sistema apresenta um comportamento de um sistema sobreamortecido. Neste caso, os pólos dominantes de malha fechada para o sistema de segunda ordem foram posicionados no plano Z sob o eixo real com valores idênticos, não apresentando componente imaginária. Uma vez que a resposta do sistema é idêntica a de um sistema sobre-amortecido, não existe a ocorrência de sobrelevação. No entanto, como já referido, a resposta transitória do sistema apresenta pequenas oscilações com amplitudes decrescentes. Este comportamento pode ser denominado de *ringing*. Isto deve-se sobretudo a um mau posicionamento dos pólos dominantes para a malha fechada. Para o segundo ensaio, posicionaram-se os dois pólos dominantes no plano Z em -0.2 e 0.2. Os pólos posicionados sobre o eixo real negativo no plano Z são denominados de *ringing poles* [42] e são responsáveis pelo efeito de oscilação apresentado anteriormente. Uma vez que o pólo em -0.2 está próximo da origem ($p=0$), o efeito de *ringing* na resposta do sistema é mínimo, tendo este um maior efeito à medida que se aproxime de -1 no plano Z [42]. De salientar que a resposta do sistema no segundo ensaio mostrou-se bastante satisfatória uma vez que, em regime estacionário, o erro apresentado foi próximo de zero, motivado pela presença de ação integral.

7.1.2.1 Introdução de um torque de carga externo

Para analisar a influência de uma carga externa no controlo de posição do sistema utilizaram-se as mesmas condições do ensaio 2 apresentado no tópico anterior no que se refere ao posicionamento de pólos e, conseqüentemente os ganhos de realimentação de estados obtidos. Considerou-se a mesma configuração da fonte de corrente utilizada no controlo de velocidade do sistema de modo a ser fornecida uma corrente constante de 250mA. De seguida, são apresentados os resultados do comportamento do sistema quando lhe é aplicado um torque carga em estado estacionário.



a) Sinais de posição e de referência



b) Sinais de posição e de referência ampliados

Figura 91 - Influência de uma carga de 250mA no controle da posição do sistema real

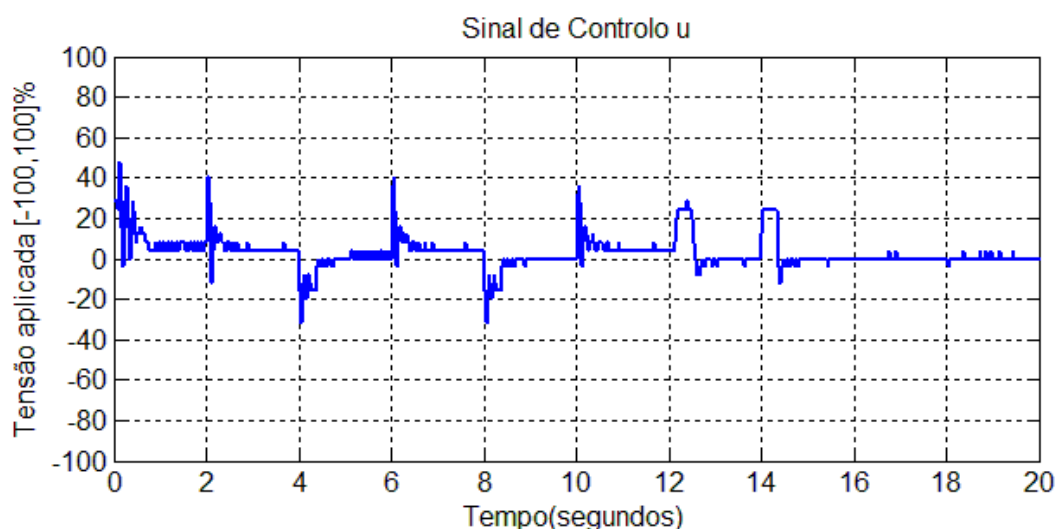


Figura 92 - Sinal de controle da posição do sistema real com influência de um torque de carga

Introduziu-se um torque a partir do motor de carga utilizando a fonte de corrente linear com a mesma configuração utilizada na seção 7.2.2.1 nos instantes de tempo $t \approx 12$ segundos e $t_1 = 14$ segundos (Figura 91 e Figura 92). Analisando o comportamento da posição do sistema real em relação à posição de referência nos instantes t e t_1 , verifica-se uma diminuição no valor da posição aproximadamente 5 graus (Figura 91 b)), aumentando assim o erro entre a posição medida pelo codificador e a posição de referência. Observa-se depois uma diminuição do erro entre a posição de referência e a posição do sistema, demorando cerca de 1 segundo a seguir a posição de referência. O sistema real apresenta um comportamento oscilatório quando reage ao torque que lhe foi aplicado externamente, por motivos apresentados na realização do ensaio 2. Relativamente ao sinal de controle, o valor de tensão aplicada ao sistema aumenta na ordem dos 25%, sendo que atinge de novo o valor zero quando o erro é corrigido.

Tal como verificado no tópico 7.2.1.1, a ação integral do sistema permite corrigir o erro em regime estacionário. Também de salientar que a correção do erro entre a posição de referência e a posição medida é feita num tempo superior ao tempo de estabelecimento definido inicialmente para a malha fechada, tornando assim mais lenta a dinâmica do sistema.

Capítulo 8 Conclusões e Trabalho Futuro

No presente capítulo, resumem-se as conclusões principais desta dissertação, referindo ainda alguns aspetos, sobre a temática abordada, que poderiam ser melhorados e/ou desenvolvidos.

Implementou-se um controlador por realimentação de estados com acessibilidade às três variáveis de estado (posição, velocidade e corrente elétrica) por meio de sensores (codificador e sensor de corrente). Com os resultados obtidos no capítulo 7, o controlador desenvolvido mostrou um bom desempenho no seguimento de referências com uma dinâmica adequada e desta forma cumprindo os requisitos definidos para a resposta transitória. Naturalmente que o desenvolvimento de um controlador de velocidade e posição com realimentação de estados para o sistema real foi precedido por diversas simulações em ambiente computacional.

O sistema operativo de tempo real (FreeRTOS) sob o qual se desenvolveu o *software* de identificação paramétrica e controlo do sistema mostrou ser de fácil utilização e compreensão e acima de tudo bem estruturado, permitindo de forma simples o desenvolvimento e implementação de tarefas e executá-las com uma periodicidade fixa cumprindo assim os requisitos temporais da tarefa.

Nesta dissertação, verificou-se que o conversor analógico-digital interno do microcontrolador LPC1759 apresentou o fenómeno de aparecimento de *glitches* durante a leitura dos valores analógicos. A solução encontrada foi a filtragem efetuada por *software*. No sentido de melhorar a medição da corrente elétrica, deveria ser utilizado um conversor analógico-digital externo ao microcontrolador.

Como complemento do trabalho desenvolvido, seria um desafio a implementação de um controlo *sensorless* do sistema real com a utilização de um Observador de Estados, sendo que o observador poderia ser de ordem total ou de ordem reduzida. Nesta dissertação, apenas foi implementado um Observador de estados de ordem completa em ambiente de simulação. De referir que a utilização de um observador de estados é uma alternativa viável à utilização de eletrónica de medição das variáveis de estado, permitindo reduzir o preço de custo do produto.

No âmbito do controlo, seria interessante desenvolver um controlador por modo de deslizamento (*Sliding Mode Controller*) do sistema eletromecânico utilizado na presente dissertação. A implementação deste controlador permitiria avaliar o desempenho do sistema por comparação com o controlador por realimentação de estados.

Por fim, verificaram-se limitações ao nível da memória do microcontrolador LPC1759 para implementar um controlador com características adaptativas, pelo que seria interessante implementar um controlador adaptativo. Assim, a utilização do algoritmo *RLS* nesta dissertação (capítulo 6) serviria de “ponte” para a implementação de um controlador adaptativo.

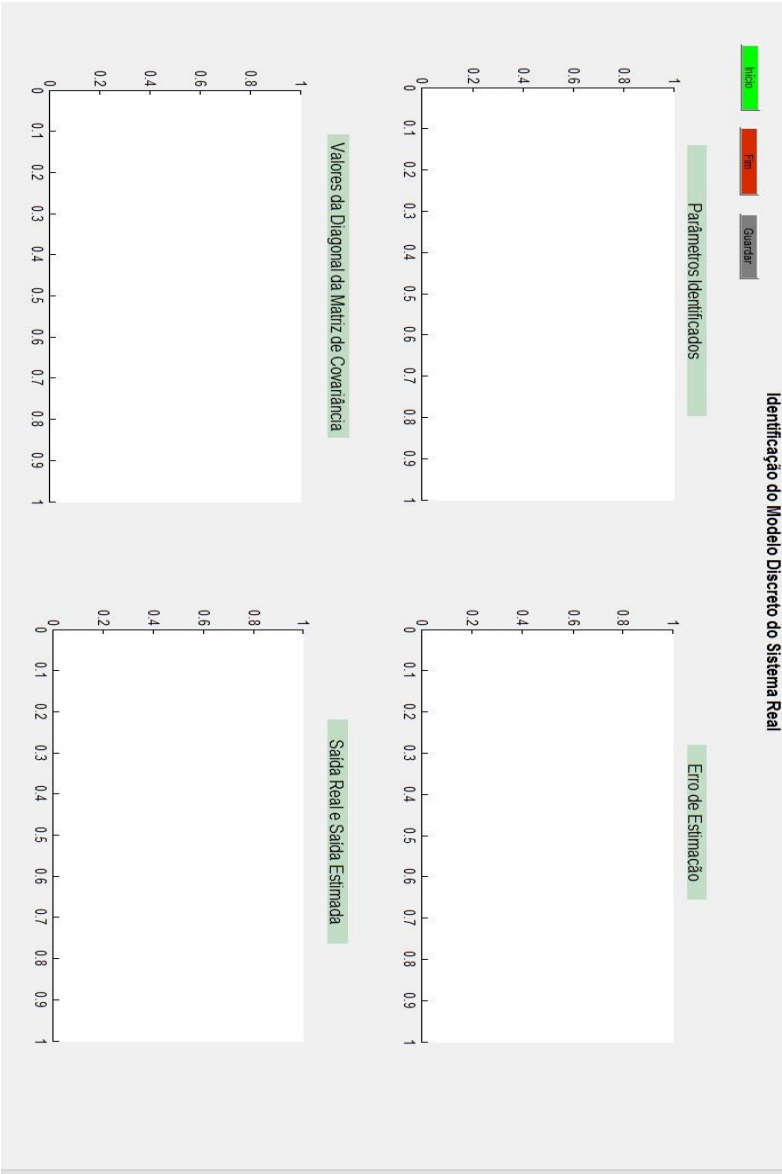
Bibliografia

- [1] <http://www.exercito.pt/sites/RAAA1>, “Boletim da Artilharia Antiaérea_2015” (acedido a 4 de Fevereiro de 2016).
- [2] Yiu Joseph, “The Definitive Guide to the ARM Cortex-M3, Second Edition”, Elsevier, 2009.
- [3] NXP Semiconductors, “LPC1759 32-bit ARM Cortex-M3 - Datasheet”, 2015.
- [4] Paulo Pedreiras, “O sistema operativo de tempo-real FreeRTOS”, UA/DETI, 2012.
- [5] Hamid A. Toliyat, “Handbook of Electric Motors”, CRC Press, 2004.
- [6] <http://www.automation-talk.info/2011/11/difference-between-pmdc-and-brushless.html> (acedido a 29 de Julho de 2015).
- [7] Luís Filipe Terra Ferreira, “Controlo Adaptativo de um Motor DC”, Universidade de Aveiro, 2010.
- [8] Lennart Ljung, “System Identification – Theory for the User”, Prentice Hall, 1987.
- [9] National Instruments, “Motor Fundamentals”, 2014.
- [10] Austin Hughes, “Electric Motors and Drives: Fundamentals, Types and Applications, 4th Edition”, Paperback, 2013.
- [11] DC Motors, <http://www.copyleftscience.wikispaces.com/9.3.1+Motors> (acedido a 27 de Julho de 2015).
- [12] Ramu Krishnan, “Electric Motor Drives: Modeling, Analysis and Control”, Prentice Hall, 2001.
- [13] Pedro Ornelas, Universidade Federal da Bahia - Escola Politécnica, “DEM, Motores Eléctricos de Corrente Continua”, Brasil, 2000.
- [14] Cristopher Kilian, “Modern Control Technology: Components and Systems”, Thomson Delamar Learning, 2001.
- [15] Daniel Thomazini, “Sensores Industriais - Fundamentos e Aplicações”, editora Érica, 2008.
- [16] Richard C. Dorf, Robert H. Bishop, “Modern Control Systems – Twelfth Edition”, Pearson Prentice Hall, 2011.

- [17] Katsuhiko Ogata, "Ingeniería de Control Moderna", tercera edición, Prentice Hall, 2010.
- [18] Benjamin C. Kuo, "Automatic Control Systems", 9th Edition, Wiley, 2009.
- [19] Katsuhiko Ogata, "Modern Control Engineering", 5th edition, Prentice Hall, 2010.
- [20] Karl J. Åström, Björn Wittenmark, "Computer-Controlled Systems: Theory and Design, Third Edition", Dover Publications, 2013.
- [21] Karl J. Åström, Björn Wittenmark, "Adaptive Control: Second Edition", Dover Publications, 2013.
- [22] Katsuhiko Ogata, "Discrete-Time Control Systems 2nd Edition", Prentice Hall, 1995.
- [23] Mathworks, "Matlab, Getting Started Guide R2011b", 2011.
- [24] Alexandre M. Mota, "Simulação de Sistemas de Controlo com MATLAB e Simulink", DETUA, 2013.
- [25] Norman S. Nise, "Control System Engineering - Sixth Edition", Wiley, 2010.
- [26] Luca Zamboni, "Getting Started with Simulink", 2013.
- [27] M. Sami Fadali, A. Visioli, "Digital Control Engineering, Second Edition: Analysis and Design", Elsevier, 2012.
- [28] Dawn Tilbury, Bill Messner, "State-Space Design Method for Motor Position Control", University of Michigan and Carnegie Mellon University, 2013.
- [29] Cadence Design Systems. "OrCAD Capture - User's Guide", 2006
- [30] Fairchild Semiconductor, "74AC245, Octal Bidirectional Transceiver with 3-STATE Inputs/Outputs", 2008.
- [31] Future Technology Devices International (FTDI), "FT232R USB UART IC Datasheet", 2015.
- [32] Intersil, "HIP4081, 80V/2.5A Peak, High Frequency Full Bridge FET Driver - Datasheet", 2015.
- [33] Allegro MicroSystems LLC, "ACS712 – Fully Integrated, Hall Effect-Based Linear Current Sensor IC with 2.1 kVRMS Isolation and a Low-Resistance Current Conductor", 2013
- [34] ON Semiconductor, "LM317 - Linear Voltage Regulator, Positive, Adjustable Output, 1.5 A".

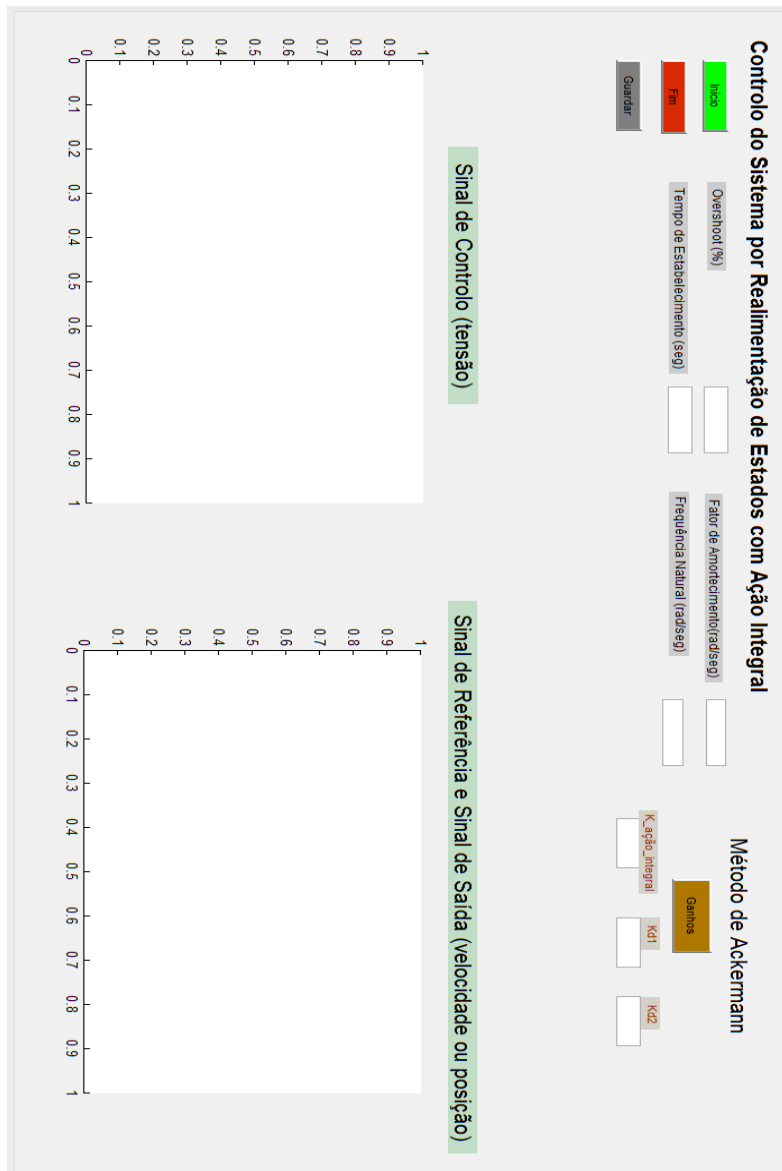
- [35] Open Source RTOS, <http://www.osrtos.com> (acedido a 17 de Agosto de 2015)
- [36] ARM-CMSIS, <http://www.arm.com> (acedido a 17 de Agosto de 2015)
- [37] ARM mbed, <https://developer.mbed.org/forum>, AnalogIn ‘glitches’ – a ‘cure’ and a tool (acedido a 04 de Dezembro de 2015)
- [38] A. Paulo, P. Costa, P. Santos, “Introdução à identificação de modelos discretos para sistemas dinâmicos, notas de ensino”, 2002.
- [39] Ana Luísa Lopes Antunes, “Algoritmos de Controlo Distribuído em Sistemas Baseados em Microprocessadores”, Universidade de Aveiro, 2008.
- [40] Alexandre M. Mota, “Identificação de Modelos Discretos através do Método dos Mínimos Quadrados”, DETUA, 2013.
- [41] Alexandre M. Mota, “Identificação de Sistemas”, DETUA, 2000.
- [42] Brian Roffel & Ben Betlem, “Advanced Practical Process Control”, Springer, 2004.

Anexo A Interface gráfica (GUI)



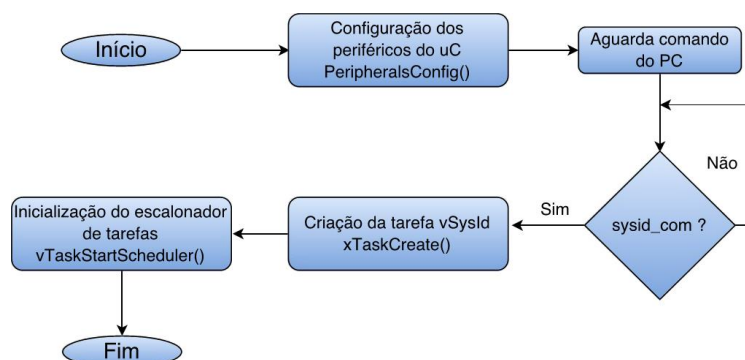
A.1 - Identificação dos parâmetros do sistema real (sem sinais)

A.2 - Controlo do sistema real sem ação integral (sem sinais)

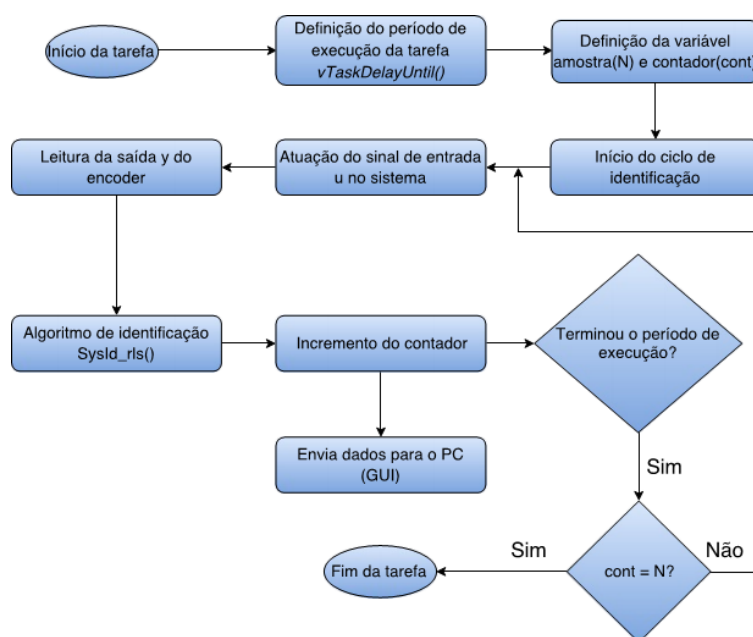


A.3 - Controlo do sistema real com ação integral (sem sinais)

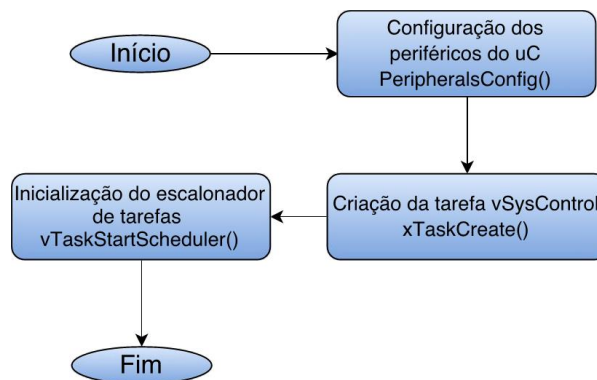
Anexo B Fluxogramas



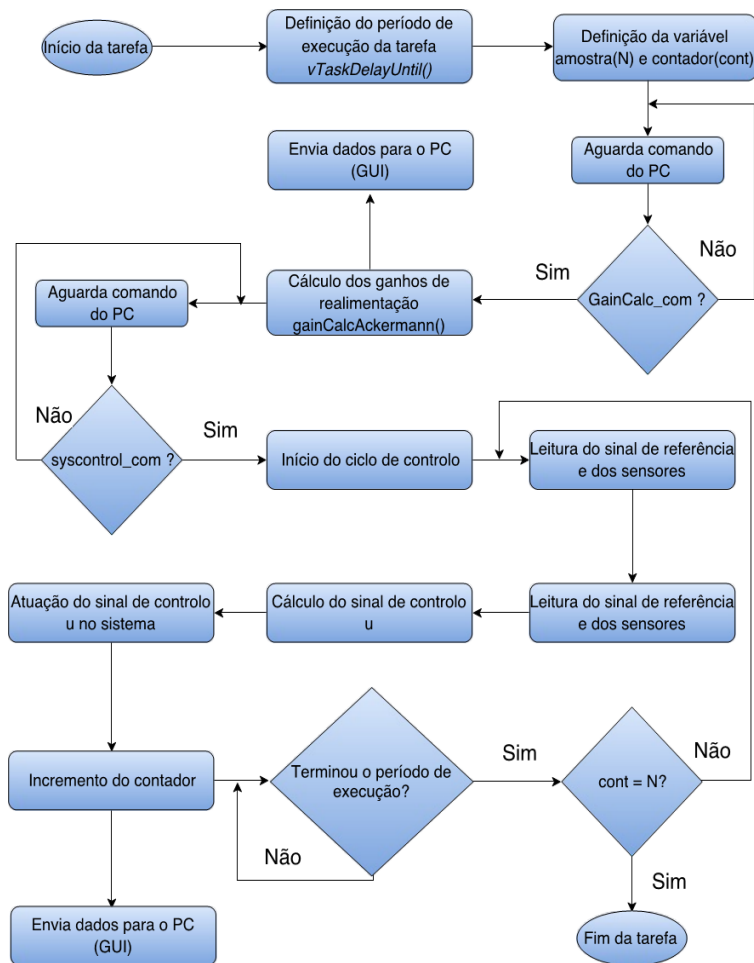
B.1A – Rotina principal do processo de identificação do sistema



B.1B - Tarefa de Identificação paramétrica do sistema real (*vSysId*)

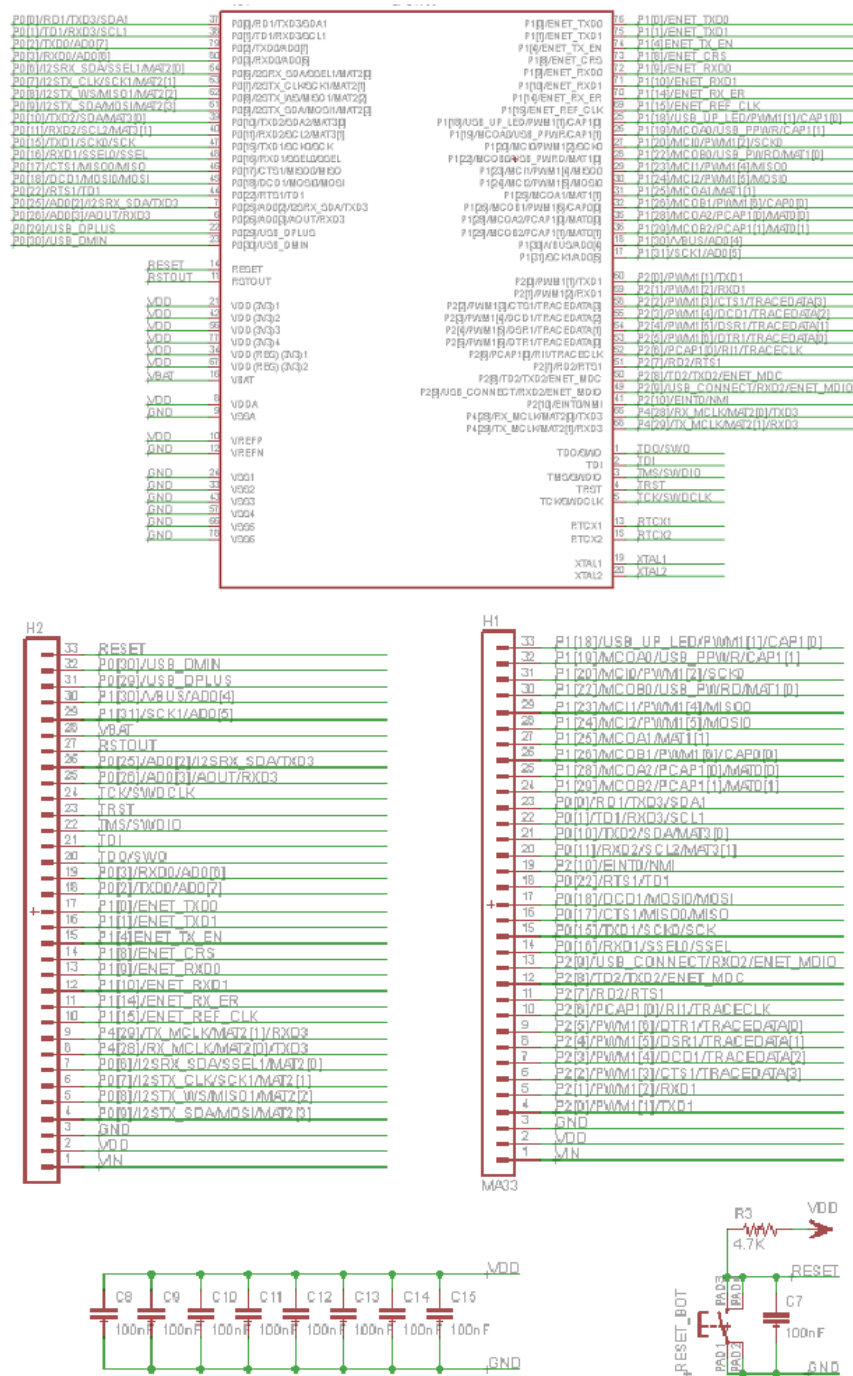


B.2A - Rotina principal do processo de controlo do sistema

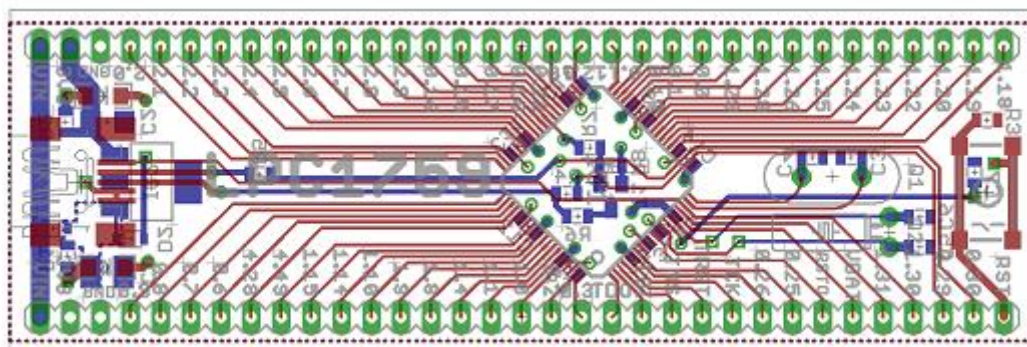


B.2B - Tarefa de controlo do sistema real (vSysControl)

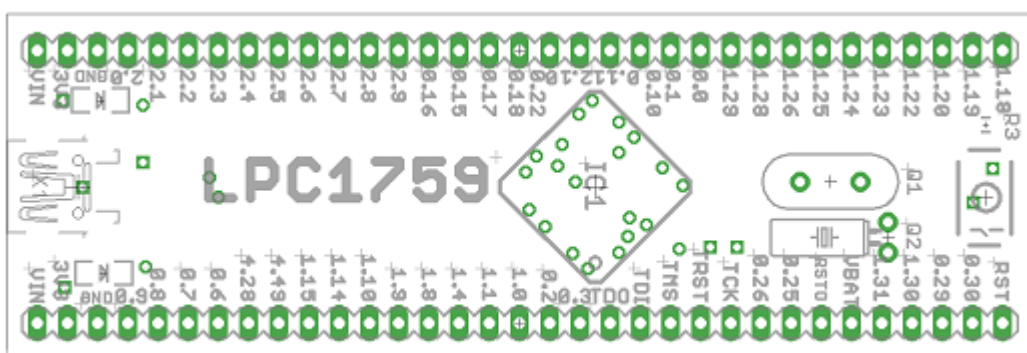
Anexo C Placa de Controle



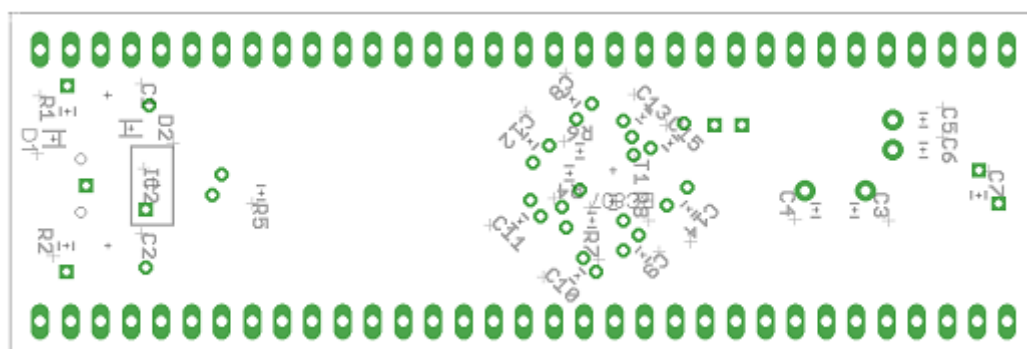
C.1 - Esquema elétrico da placa LPC1759



C.2 - Layout geral da placa LPC1759

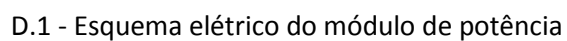


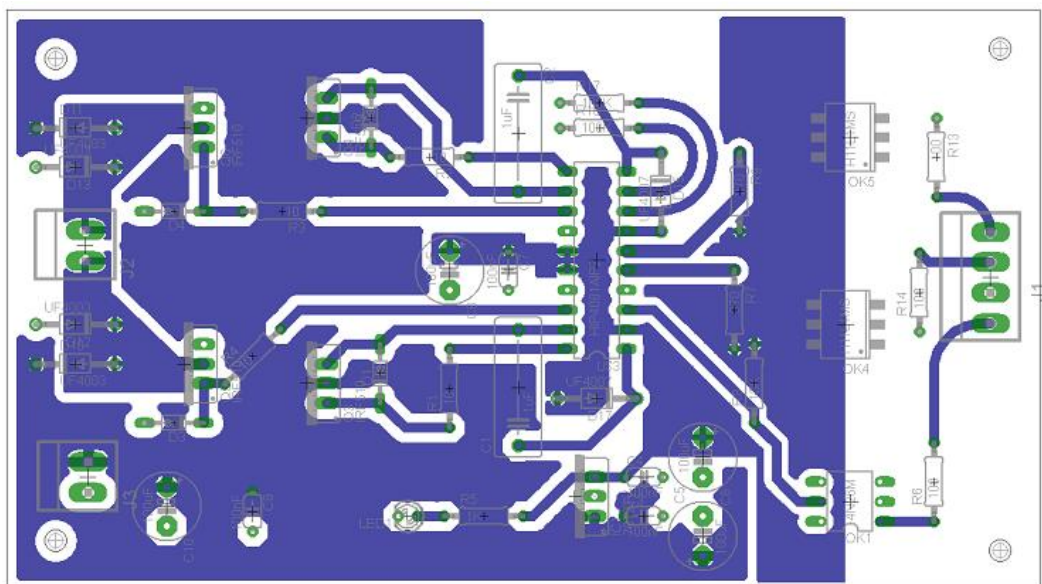
C.3 - Layout superior da placa LPC1759



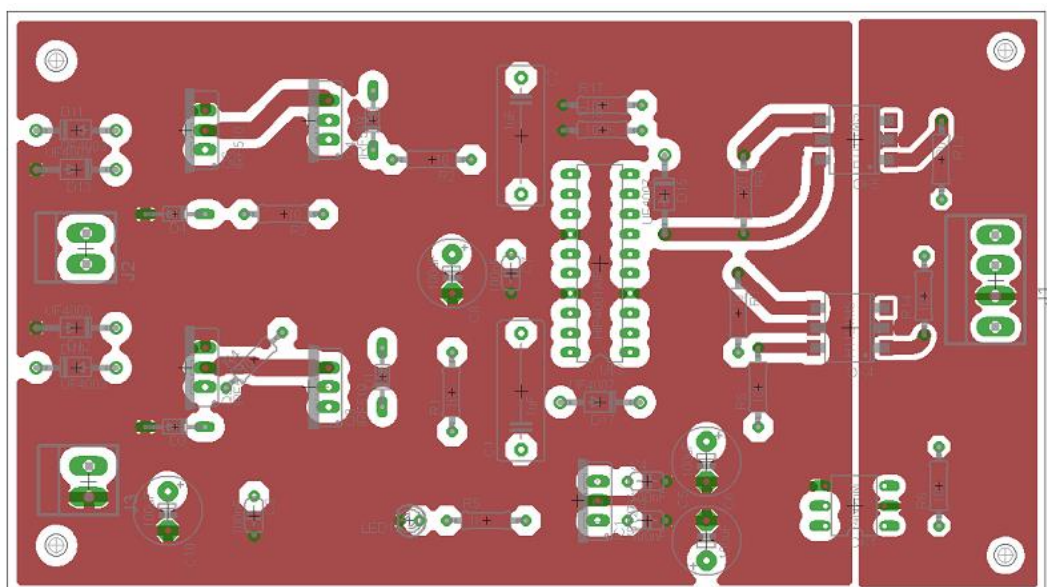
C.4 - Layout inferior da placa LPC1759

122





D.2 - Layout superior do módulo de potência



D.3 - Layout inferior do módulo de potência